



Universitat de València

**FACULTAD DE FILOSOFÍA Y CIENCIAS DE LA
EDUCACIÓN**

**PROGRAMA DE DOCTORADO EN EDUCACIÓN
(RD 99/2011)**

**Diseño de un Modelo Pedagógico basado en
Computación en la Nube y Computación Afectiva
para el desarrollo de las habilidades de
programación**

TESIS DOCTORAL

Presentada por: Pablo Alejandro Quezada Sarmiento

Director: Dr. Cristóbal Nico Suárez Guerrero

Tutor: Dr. Cristóbal Nico Suárez Guerrero

Junio – 2025



Universitat de València

**FACULTAD DE FILOSOFÍA Y CIENCIAS DE LA
EDUCACIÓN**

**PROGRAMA DE DOCTORADO EN EDUCACIÓN
(RD 99/2011)**

**Diseño de un Modelo Pedagógico basado en
Computación en la Nube y Computación Afectiva
para el desarrollo de las habilidades de
programación.**

TESIS DOCTORAL

Presentada por: Pablo Alejandro Quezada Sarmiento

Director: Dr. Cristóbal Nico Suárez Guerrero

Tutor: Dr. Cristóbal Nico Suárez Guerrero

Junio – 2025

D/ **Dr. Cristóbal Nico Suárez Guerrero**, Titular/ del Dpto. de Educación de la Universitat de València

CERTIFICA/N:

Que la presente memoria, titulada “**Diseño de un Modelo Pedagógico basado en Computación en la Nube y Computación Afectiva para el desarrollo de las habilidades de programación**”, corresponde al trabajo realizado bajo mi dirección por D/ Pablo Alejandro Quezada Sarmiento, para su presentación como Tesis Doctoral en el Programa de Doctorado en Educación de la Universitat de València.

Y, para que conste firmo, el presente certificado en Valencia, a 9 de abril 2025

Fdo.

**CRISTOBAL
NICO|SUAREZ|
GUERRERO**

Firmado digitalmente
por CRISTOBAL NICO|
SUAREZ|GUERRERO
Fecha: 2025.04.10
08:25:00 +02'00'

Dr. Cristóbal Nico Suárez Guerrero

Director de Tesis

ÍNDICE

Índice de Tablas	v
Índice de Figuras.....	vi
Agradecimientos	viii
Resumen	ix
Introducción	xv
Capítulo 1: Planteamiento del problema de la investigación.....	1
1.1 Motivación de la investigación	2
1.2 Importancia del tema.....	11
1.3 Contexto de la investigación.....	17
1.3.1 Preguntas de investigación de la tesis doctoral	18
1.4 Objetivos de la tesis doctoral	19
1.5 Metodología de la investigación.....	20
1.6 Contribución de la investigación propuesta.....	30
1.6.1 Artículos en revistas indexadas	30
1.6.2 Capítulos de libro.....	31
1.6.3 Artículos publicados e indexados en eventos científicos	33
Capítulo 2: La dimensión tecnológica.....	36
2.1 Computación en la Nube (CN)	39
2.1.1 Conceptualizaciones de CN.....	39
2.2 Tipos de Computación en la Nube.....	44
2.2.1 Tipos de CN según la propiedad del servicio	45
2.2.2 Según el modelo de servicio	47
2.3 Soluciones tecnológicas actuales de CN	49
2.3.1 Amazon Web Services.....	50
2.3.2 Google Cloud Platform.....	50
2.3.3 Microsoft Azure	50
2.3.4 Docker.....	51
2.3.5 YellowCircle	53
2.4 Computación de la nube en la educación.....	53
2.5 Herramientas relevantes de Computación en la Nube utilizadas en el contexto de la enseñanza de la Programación en Educación Superior	55
2.5.1 Alfresco	55
2.5.2 Basecamp	58

2.5.3 Edmodo	59
2.5.4 Redbooth.....	59
2.5.5 DotProject.....	59
2.5.6 BITRIX24.....	60
2.5.7 Redmine	61
2.5.8 Egroupware	62
2.5.8 Onlyoffice Formerly TeamLab Office.....	63
2.5.9 Open Atrium	63
2.5.10 Project Open	63
2.5.11 Jira.....	64
2.5.12 Stormboard	65
2.5.13 Project2manage	65
2.5.14 Teamwork.com.....	65
2.5.15 Diigo	66
2.5.16. Editores de código online	66
2.6 Computación Afectiva.....	67
2.6.1 Definiciones de Computación Afectiva (CA)	67
2.6.2 Computación afectiva en la educación.....	70
2.6.3 Modelos de CA	72
2.6.4 Consideraciones teóricas acerca de la CA en el proceso de enseñanza-aprendizaje de la Educación Superior	74
2.7 El valor formativo de la CN y la CA	76
2.8 Aspectos éticos y Seguridad de Datos en CN.....	77
Capítulo tres: La dimensión pedagógica.....	79
3.1. La lectura pedagógica del uso educativo de la tecnología.....	80
3.2 Análisis pedagógico de la educación con tecnología en pandemia	83
3.3 Definiciones de modelo pedagógico.....	86
3.4. Tipos de modelos pedagógicos	88
3.5. Modelos Pedagógicos con uso de tecnología	90
3.6 Estado del arte sobre los modelos pedagógicos con tecnología en educación superior	92
3.7 Contexto de la educación superior en la enseñanza de la programación	101
3.8 Plataformas digitales en la educación superior.....	108
3.9. Competencias digitales	110
3.10 Dimensiones de la pedagogía digital.....	114
Capítulo cuatro: La enseñanza superior de la computación en Ecuador.....	123

4.1. Contexto de las Ciencias de la Computación e Informática en el Ecuador	124
4.2. Instituciones de educación superior participantes en los casos cuasiexperimentales	131
4.2.1. Universidad Técnica Particular de Loja	131
4.2.2. Universidad Internacional del Ecuador	136
4.2.3. Universidad Nacional de Loja	138
Capítulo cinco: Metodología de la investigación	143
5.1 Método de investigación Diseño de la Ciencia	144
5.1.1 Implementación de la metodología propuesta Diseño de la Ciencia	144
5.2 Revisión sistemática de la literatura	150
5.2.1 Planificación de la revisión	151
5.2.2 Desarrollo de la revisión	156
5.2.3 Informar la revisión	158
5.3 Casos Cuasi experimentales	160
5.4 Instrumentos utilizados	163
5.4 .1 Observación	163
5.4.2 Entrevista	167
Capítulo 6: Resultados	174
6.1 Desarrollo de los escenarios cuasi experimentales	175
6.1.1 Caso Cuasi experimental 1: Programación Orientada a Objetos (POO) sin uso de CN y CA (Universidad Internacional del Ecuador -UIDE)	176
6.1.1.2 Caso cuasi experimental 1.2: POO con uso de CN Y CA (Universidad Internacional del Ecuador)	179
6.1.2 Caso cuasiexperimental.2: Uso de herramientas de CN GitHub en la Universidad Nacional de Loja.	186
6.1.2.1 Descripción del repositorio utilizado	187
6.1.3 Caso cuasi experimental 3. Prototipos Web Desarrollados mediante herramientas de CN (Universidad Técnica Particular de Loja)	194
6.1.3.1 Análisis y Planificación del Sitio Web desarrollado mediante el soporte de herramientas de CN.	196
6.1.4 Caso Cuasi experimental 4 Desarrollo de aplicativo Web y Móvil (Universidad Técnica Particular de Loja)	198
6.1.5 Caso Cuasi experimental 5 Desarrollo de aplicativo web y móvil: tienda solidaria “Everything shop” (Universidad Nacional de Loja)	206
6.1.6 Caso Cuasi experimental 6: Desarrollo de Aplicativo Móvil para la Gestión de Restaurantes (Universidad Internacional del Ecuador)	215
Capítulo7: Desarrollo de modelo pedagógico	222

7.1 Pedagogía en la educación superior.....	223
7.2 Modelo Pedagógico Propuesto	224
Conclusiones	236
Referencias.....	241
Anexos.....	272
Anexo 1: Ficha de Observación de Clase	273
Anexo 2:Entrevista para Clases de Programación	276
Anexo 3: Evaluación a nivel de Código	278

Índice de Tablas

Tabla 1	<i>Características de los casos cuasi experimentales</i>	28
Tabla 2	<i>Resumen de Producción Científica generada durante el desarrollo de la tesis doctoral</i>	35
Tabla 3	<i>Características de modelos pedagógicos</i>	89
Tabla 4	<i>Características del E-learning</i>	91
Tabla 5	<i>Características de la gestión de la calidad educativa</i>	103
Tabla 6	<i>Características proceso de enseñanza–aprendizaje</i>	107
Tabla 7	<i>Universidades Ecuatorianas que ofertan carreras relacionadas a las Ciencias de la Computación</i>	126
Tabla 8	<i>Características de la Carrera de Computación de la UTPL</i>	133
Tabla 9	<i>Hitos de la UTPL</i>	134
Tabla 10	<i>Descripción del perfil de la carrera de Informática y Multimedia</i>	137
Tabla 11	<i>Hitos de la UIDE</i>	137
Tabla 12	<i>Hitos de la Universidad Nacional de Loja</i>	141
Tabla 13	<i>Síntesis de los casos cuasiexperimentales</i>	145
Tabla 14	<i>Fases de la revisión sistemática de la literatura</i>	151
Tabla 15	<i>Cloud computing on teaching programming</i>	153
Tabla 16	<i>Cloud computing and affective computing on teaching programming</i>	153
Tabla 17	<i>Detalles de los casos cuasi experimentales</i>	160
Tabla 18	<i>Registro fotográfico de desarrollo de casos cuasiexperimentales</i>	164
Tabla 19	<i>Ventajas de Alfresco</i>	181
Tabla 20	<i>Características de las Herramientas Cloud utilizadas</i>	184
Tabla 21	<i>Características de las Herramientas Cloud utilizadas en los proyectos de software</i>	184
Tabla 22	<i>Interactividad y Retroalimentación generada</i>	185
Tabla 23	<i>Resultados del uso de las herramientas de CN</i>	186
Tabla 24	<i>Elementos GitHub</i>	187
Tabla 25	<i>Actividades propuestas en la Metodología de DC</i>	195
Tabla 26	<i>Fases de las Iteraciones</i>	200
Tabla 27	<i>Fases de Mobile D</i>	200
Tabla 28	<i>Principios de LEAN</i>	210
Tabla 29	<i>Roles LEAN</i>	211

Índice de Figuras

Figura 1	<i>Fase de diseño de la ciencia</i>	21
Figura 2	<i>Planificación de la Revisión</i>	23
Figura 3	<i>Conducción de la Revisión</i>	23
Figura 4	<i>Reporte de la Revisión</i>	24
Figura 5	<i>Resumen de casos cuasiexperimentales</i>	28
Figura 6	<i>Docker versión escritorio</i>	52
Figura 7	<i>Docker versión Online</i>	53
Figura 8	<i>Características de la herramienta de la nube Alfresco</i>	56
Figura 9	<i>Interfaz de Alfresco en Proyecto de Software</i>	56
Figura 10	<i>Interfaz Bitrix</i>	61
Figura 11	<i>Características de la herramienta de computación en la nube REDMINE</i>	62
Figura 12	<i>Características de EGROPPWARE</i>	62
Figura 13	<i>Características de la Herramienta Project Open</i>	64
Figura 14	<i>Consulta en base de datos Scopus</i>	93
Figura 15	<i>Consulta de base de datos WOS</i>	93
Figura 16	<i>Consulta de base de Datos IEEE</i>	94
Figura 17	<i>Consulta en base de datos ACM</i>	94
Figura 18	<i>Consulta Scielo</i>	95
Figura 19	<i>Consulta de base datos Scopus</i>	95
Figura 20	<i>Consulta en Base de Datos Scopus</i>	96
Figura 21	<i>Consulta en WOS</i>	96
Figura 22	<i>Consulta en la base de datos IEEE</i>	97
Figura 23	<i>Consulta ACM Pregunta de Investigación en inglés</i>	97
Figura 24	<i>Consulta ACM Pregunta de Investigación en español</i>	98
Figura 25	<i>Consulta base de datos Scielo</i>	98
Figura 26	<i>Planificación de la revisión sistemática de la literatura</i>	151
Figura 27	<i>Paso 1 de la planificación de la revisión</i>	152
Figura 28	<i>Paso 2 de la planificación de la revisión sistemática de la literatura</i>	154
Figura 29	<i>Criterios de inclusión y exclusión de los estudios primarios</i>	155
Figura 30	<i>Fases del desarrollo del protocolo de la revisión sistemática de la literatura</i>	158
Figura 31	<i>Reporte de la revisión</i>	159
Figura 32	<i>Moodle POO</i>	177
Figura 33	<i>Grupo de Control</i>	178
Figura 34	<i>Grupo de Control 2</i>	180
Figura 35	<i>Sitio Alfresco POO</i>	181
Figura 36	<i>Red Social Generada en Redbooth</i>	182
Figura 37	<i>Grupo de Control</i>	187
Figura 38	<i>Repositorio GutHub Computación en la Nube en Marco del Proyecto de la tesis doctoral</i>	188
Figura 39	<i>Herramientas de CN GitHub</i>	190
Figura 40	<i>Repositorio Implementado</i>	190
Figura 41	<i>Repositorio de programas desarrollados</i>	191
Figura 42	<i>Retroalimentación de proyectos</i>	192
Figura 43	<i>Control de Proyectos</i>	192
Figura 44	<i>Grupo de Control cuasiexperimental</i>	194
Figura 45	<i>Sitio Web desarrollado</i>	197

Figura 46	<i>Grupo de Control</i>	199
Figura 47	<i>Interfaces de Aplicación Móvil</i>	202
Figura 48	<i>Sitio web desarrollado en el caso cuasi experimental</i>	203
Figura 49	Configuración del ambiente de desarrollo.....	203
Figura 50	<i>Interfaz de la aplicación desarrollada</i>	205
Figura 51	<i>Tablero Kanban de Tobeio</i>	205
Figura 52	<i>Estudiantes participantes del grupo de control 5</i>	207
Figura 53	<i>Aplicación Web Desarrollada</i>	212
Figura 54		213
Figura 55	Prototipo Desarrollado.....	214
Figura 56	Registro de emociones y generación de resultados de análisis emocional	214
Figura 57	<i>Grupo de Control</i>	216
Figura 58	<i>Interfaz de la aplicación</i>	218
Figura 59	Interfaz de Desarrollo de la aplicación móvil	218
Figura 60	<i>Menú del aplicativo y api de google maps</i>	219
Figura 61	<i>Funcionalidades implementadas</i>	219

Agradecimientos

Mi eterno agradecimiento a la Universitat de València, en particular al Programa de Doctorado en Educación, por la formación profesional brindada en el desarrollo de la presente tesis doctoral; a mi director y tutor de tesis Dr. Cristóbal Nico Suárez Guerrero, por sus sabias orientaciones y apoyo constante en el desarrollo de este trabajo y por haber publicado en congresos de relevancia y revistas indexadas de alto impacto el proceso y resultados de esta investigación.

Un especial agradecimiento a mis padres: Narcisa Mariana de Jesús Sarmiento Ochoa Walter Quezada Ambuludi mi hermanita querida Narcisa Mariana Quezada Sarmiento, por ser el motivo para superarme cada día y por el constante apoyo y cariño que me brindan. A mi querida esposa Patricia Marisol Chango Cañaveral, a mi amado hijo Pablo Andrés y a mi hija Renata Emilia Quezada Chango por ser el motor de mi vida y estar presentes en el cumplimiento de esta meta.

Resumen

El modelo pedagógico propuesto en esta investigación combina tecnologías emergentes como la Computación en la Nube (CN) y la Computación Afectiva (CA) con la finalidad de potenciar el desarrollo de habilidades de programación en estudiantes universitarios. Esta tesis doctoral, por tanto, busca hacer una lectura pedagógica de las CN y la CA desde un modelo, examinado en la parte empírica, centrado en el estudiante para promover el aprendizaje activo, la personalización y la interacción constante entre el docente y los alumnos.

Es importante manifestar que las herramientas de CN, dentro del desarrollo de la presente investigación, proporcionó una infraestructura escalable y accesible que permitió a los estudiantes trabajar en entornos de desarrollo en línea sin restricciones físicas. Así mismo la CN, facilitó el acceso a recursos educativos como plataformas de codificación, simuladores y repositorios de aprendizaje, mejorando la experiencia práctica. Por otro lado, la CA permitió integrar sistemas que analizan las emociones y estados afectivos de los estudiantes, lo que hizo posible ajustar el grado de dificultad de las actividades y ofrecer retroalimentación ajustada a cada alumno, en tiempo real, aumentando la motivación y reduciendo la ansiedad. Esto es, se trató de promover un aporte pedagógico desde las tecnológicas de la CN y la CA.

En este contexto es importante considerar que esta tesis puede formar parte de los desarrollos que la pedagogía digital promueve y, así, entender y orientar mejor las prácticas formativas universitarias con tecnología emergente, como las analizadas aquí. Aunque esta tesis doctoral se despliega sobre la formación universitaria en computación, el aporte gira en torno a las estrategias y los modelos educativos que pueden y deben mejorar la experiencia de aprendizaje. Esta tesis, como otras en esta línea, buscan contribuir con el desarrollo de sistemas, planes y programas de las Instituciones de Educación Superior (IES) y, con ello, buscan generar un ecosistema digital para la enseñanza en las distintas modalidades de

estudio. Es más, en un mundo donde la tecnología facilita el acceso a la educación abriendo oportunidades al desarrollo profesional, tanto de docentes como de estudiantes, hace falta investigar para lograr que estas nuevas tecnologías sean realmente importantes para el quehacer educativo. Esta tesis busca ser parte de esta idea global.

Esta investigación surgió también de la necesidad experimentada en el COVID-19 donde se puso en evidencia las profundas deficiencias de los modelos pedagógicos virtuales. Las IES, a nivel mundial, vieron con nuevos ojos la importancia de una adecuada pedagogía digital que fortalezca las capacidades de los docentes y estudiantes en el uso de la tecnología y, así, el uso de nuevas herramientas basadas en la CN y CA en el proceso de enseñanza y aprendizaje. Así, el objetivo general de esta tesis doctoral, fue Diseñar un Modelo Pedagógico basado en herramientas de CN y CA para el desarrollo de las habilidades de programación en el área informática. Para el logro de este gran objetivo, se desarrollaron de forma específica los siguientes objetivos:

- Conocer el estado del arte referente al uso de herramientas de CN y la relación de la CA en el desarrollo de las habilidades de programación en el área informática.
- Analizar el uso herramientas de CN y la relación de la CA para la mejora de las habilidades de programación en alumnos del área de informática que cursaron asignaturas con el componente de programación de tres universidades ecuatorianas.
- Valorar y establecer las estrategias metodológicas para la enseñanza de programación en el área de informática en el ámbito de la educación superior.
- Articular la CN y la CA en un modelo de aprendizaje que favorezca el aprendizaje de la programación en el contexto de ingeniería informática.

Para lograr el objetivo, el diseño pedagógico, se emplearon dos estrategias metodológicas de investigación clave.

Por un lado, para dar cuenta del estado del arte referente al uso de la CN y la relación de la CA en el desarrollo de las habilidades de programación en el área Informática se realizó una revisión sistemática. Este análisis permitió identificar tendencias, desafíos y oportunidades en su uso dentro del ámbito educativo, así como las mejores prácticas para su implementación. La revisión también sirvió como base para justificar la pertinencia de integrar estas tecnologías emergentes en el desarrollo de habilidades de programación, asegurando que la propuesta estuviera alineada con las necesidades actuales y futuras del contexto educativo.

La parte de resultados de la presente investigación, se desarrolló a través de la exploración en seis casos con la participación de 139 estudiantes del área de ciencias de la computación, de tres universidades de Ecuador, bajo la metodología del Diseño de la Ciencia en adelante (DC). En estos seis casos se emplearon aplicaciones web y móviles, para conectar la teoría con la práctica, ya que el modelo pedagógico no solo buscó desarrollar competencias técnicas relacionadas con las carreras, sino también promover habilidades blandas como el trabajo en equipo, la comunicación y la gestión emocional, aspectos clave para los programadores modernos. En estas experiencias analizadas se propusieron estrategias metodológicas para la enseñanza de programación en el área de informática en la educación superior desde propuestas como las clases invertidas, la gamificación y la programación en pares para fomentar la colaboración, la resolución de problemas y el aprendizaje autónomo basados en la CN y la CA. El DC permitió estructurar el proceso de investigación de manera rigurosa y sistemática. Este enfoque se aplicó en todas sus fases: desde la identificación del problema y el análisis del contexto, hasta el diseño, implementación, evaluación y refinamiento del modelo propuesto. Cada etapa se llevó a cabo bajo criterios metodológicos precisos, garantizando la validez y aplicabilidad de los resultados obtenidos.

La combinación DC y la revisión sistemática de la literatura no solo fortaleció el rigor del proceso investigativo, sino que también permitió construir un modelo pedagógico

fundamentado en evidencia, capaz de adaptarse a los entornos dinámicos y a las demandas del aprendizaje contemporáneo.

Así, la presente tesis doctoral se estructura en siete capítulos que abarcan los temas relacionados con el uso de la CN y CA, como herramientas tecnológicas y pedagógicas para desarrollar un modelo pedagógico que contribuya a mejorar las habilidades de programación de los estudiantes del área de ingeniería informática. Y la final se presentan las conclusiones del trabajo. También dispone de una sección de anexos donde se puede ver los instrumentos de investigación empleados.

En el capítulo 1 se presenta el planteamiento del problema de investigación, en el capítulo 2 se ofrece una mirada a la dimensión tecnológica, el capítulo 3 se enfoca en la dimensión pedagógica, el capítulo 4 muestra información referente a la enseñanza de la programación en la educación superior en Ecuador, el capítulo 5 muestra la metodología de investigación que soporta la tesis y los resultados de los casos cuasiexperimentales desarrollados son evidenciados en el capítulo 6. El producto, es decir, el modelo pedagógico basado en CN y CA, se presenta en el capítulo 7 junto a las conclusiones respectivas. Finalmente, las referencias bibliográficas son mostradas como soporte científico de la tesis doctoral.

Los procesos educativos y pedagógicos digitales, que se desarrollaron en la presente tesis doctoral, se basaron en metodologías activas de aprendizaje y actividades didácticas que permitieron a los participantes construir conocimiento y sentido a las situaciones que viven en el proceso educativo y a los docentes facilitar el aprendizaje, mediado por las herramientas de CN y CA, para mejorar las habilidades programación.

Gracias a la revisión sistemática de la literatura se puede concluir que la CN facilita el acceso a entornos de programación flexibles y escalables, reduciendo las barreras tecnológicas y promoviendo un aprendizaje colaborativo. Por otro lado, la CA contribuye significativamente al aprendizaje adaptativo al permitir la personalización de contenidos en

función de las emociones y respuestas cognitivas del estudiante. El análisis de la literatura sugiere que los modelos pedagógicos tradicionales no siempre son efectivos para la enseñanza de la programación, especialmente en contextos donde los estudiantes enfrentan dificultades relacionadas con la abstracción y resolución de problemas de programación. En este sentido, la combinación de CN y CA ofrece una oportunidad innovadora para mejorar la motivación, la retención del conocimiento y la autonomía de los estudiantes participantes en los casos cuasiexperimentales desarrollado.

De forma concreta, gracias al estudio de los seis casos analizados, este trabajo evidenció que el uso de herramientas basadas en CN fomentó un entorno de aprendizaje más interactivo entre docentes y estudiantes. Se establecieron redes de colaboración efectivas, lo que incrementó la participación y la interacción entre los estudiantes. No obstante, aunque estas herramientas ofrecen múltiples ventajas, su adopción plena y efectiva requiere abordar de manera adecuada los problemas de seguridad y protección de datos inherentes a la CN.

La integración de tecnologías basadas en CN en los currículos de Ciencias de la Computación ha demostrado ser una tecnología efectiva para aumentar la motivación de los estudiantes, fomentar el trabajo en equipo y mejorar la eficiencia y productividad en el desarrollo de código. La integración de estas tecnologías en modelos pedagógicos ofrece entornos que facilitan la interacción con la información y la creación de contenido, fortaleciendo así el aprendizaje práctico y colaborativo.

La CA ha emergido como un enfoque innovador en la enseñanza de la programación, al abordar los desafíos emocionales y cognitivos que enfrentan los estudiantes en el proceso de aprendizaje. La literatura revisada evidencia que las emociones juegan un papel fundamental en la adquisición de habilidades de programación, influyendo directamente en la motivación, la persistencia y la capacidad de resolución de problemas.

Uno de los principales beneficios de la CA fue la capacidad para personalizar el proceso de enseñanza, ajustando el ritmo y la dificultad de las tareas en función del estado

emocional del estudiante que participo en los casos cuasiexperimentales. La CA representa un avance significativo en la enseñanza de la programación, al ofrecer un enfoque centrado en el estudiante que mejora la motivación y el rendimiento académico.

No obstante, durante el desarrollo de esta investigación se identificaron diversas limitaciones que pueden afectar la implementación del modelo pedagógico basado en CN y CA propuesto:

1. Limitaciones tecnológicas: Dependencia de la conectividad a internet, infraestructura insuficiente o restricciones en el acceso a plataformas en la nube.
2. Falta de formación docente en pedagogía digital: Necesidad de capacitación en metodologías de enseñanza apoyadas en CN y CA.
3. Aspectos éticos en el uso de herramientas de CN y CA: Posibles sesgos algorítmicos, uso indebido de datos y dependencia de plataformas comerciales.
4. Falta de actualización curricular: Incorporación insuficiente de tecnologías emergentes en los planes de estudio de Ciencias de la Computación.
5. Riesgos de privacidad y protección de datos: Preocupaciones sobre la recopilación, almacenamiento y manejo de información sensible de los estudiantes.

El diseño de un modelo pedagógico basado en CN y CA representa una oportunidad significativa para la enseñanza de la programación. No obstante, su implementación efectiva depende de la capacidad para mitigar los desafíos mencionados, garantizando un equilibrio entre innovación tecnológica y seguridad educativa.

Introducción

De acuerdo a Slater et al. (2010) “ Los entornos educativos en el ámbito de educación superior se han visto sometidos en las últimas décadas a una mayor presión por la competitividad global”(p.5). La mayor exigencia de la sociedad a los agentes sociales y de desarrollo, en especial en el contexto de la formación de ingenieros informáticos, es que sus habilidades estén alineadas al desarrollo de la tecnología y su aporte a la sociedad (Gumusluoglu & Ilsev, 2009).

La CN también proporcionó a estudiantes y profesores una plataforma flexible para desplegar recursos informáticos bajo demanda, ajustándose a las necesidades específicas del proceso de enseñanza-aprendizaje. Por ejemplo, los docentes pueden crear máquinas virtuales con software preinstalado para implementar rápidamente laboratorios de programación, optimizando el tiempo y mejorando la experiencia educativa.

Las Tecnologías de Información y Comunicación en adelante (TIC) generan nuevos lenguajes y formas de representación y facilitan la creación de nuevos escenarios de aprendizaje. Por ello, las instituciones educativas no pueden permanecer al margen, deben conocer y utilizar estos nuevos lenguajes y formas de comunicación (Barreto & Diaz, 2017). Scherer (2009) manifiesta que las emociones son fenómenos multifacéticos que involucran procesos psicológicos coordinados que incluyen aspectos afectivos, cognitivos, fisiológicos, motivacionales y expresivos. Por ejemplo, cuando un estudiante de ingeniería informática tiene que codificar un programa o aplicación, en un tiempo determinado, puede estar muy nervioso (afecto), inseguro de sus conocimientos (cognición), su presión arterial puede ser alta (fisiología), se siente frustrado por el último examen (motivacional) y su rostro y voz denotan preocupación (expresión).

Numerosas investigaciones, como las de Zhang et al. (2021) han demostrado que un estado emocional positivo fomenta enfoques más creativos para resolver problemas, particularmente en áreas técnicas como la informática. Este hallazgo ha llevado a los

científicos del área a desarrollar sistemas capaces de reconocer automáticamente las emociones mediante técnicas de inteligencia artificial y reconocimiento de patrones.

El reconocimiento automático de emociones puede mejorar el desempeño, la usabilidad y la calidad de sistemas de software que interactúan con el ser humano, además de los resultados de los ambientes virtuales de aprendizaje, por ello, se tomó como referencia los aspectos relevantes de la CA que combinados con el uso de las herramientas de CN son la base para la propuesta del modelo pedagógico desarrollado en el marco de esta tesis doctoral. Pues bien, la idea es unir ambos potenciales tecnológicos (CN y CA) en un diseño pedagógico.

De acuerdo con Quezada y Suárez (2021) las nuevas tecnologías y en especial los paradigmas computacionales pueden ser empleados en diversos campos, entre ellos el educativo. Este hecho permite el diseño, búsqueda, presentación, intercambio y reutilización de objetos de aprendizaje, dado que las tecnologías hacen posible almacenar, organizar, replicar, difundir y transformar, lo que implica ahorro de tiempo y recursos educativos. Esta tesis doctoral se enfocó en el Diseño de un Modelo Pedagógico basado en CN y CA para el desarrollo de las habilidades de programación en el área de la Ingeniería Informática, con ello se propone aportar en el área educativa y al perfil profesional de los estudiantes de ingeniería en informática.

Capítulo 1: Planteamiento del problema de la investigación

1.1 Motivación de la investigación

León (2007) manifiesta que “la educación es un proceso sumamente complejo y no se reduce a la enseñanza. Para definirla es necesario considerar la condición y naturaleza humana, además de la cultura en su conjunto” (p.3).

De acuerdo con Jornet y Sancho (2018) la educación es un factor que mejora el desarrollo personal y la transformación social. Se observa, a partir de diversos indicadores sociales, que las personas con un grado educativo superior presentan mayores niveles socioeconómicos y menor probabilidad de exclusión social a lo largo de la vida. Por ello la educación no solo tiene un impacto en el desarrollo personal de los individuos, sino también en la transformación social. La educación puede actuar como un motor para el cambio social al empoderar a las personas con conocimientos, habilidades y perspectivas que les permiten contribuir de manera significativa a sus comunidades y a la sociedad en general.

Merchesi (2000) argumenta que: “Los indicadores sociales que muestran una correlación entre niveles educativos más altos y mejores condiciones socioeconómicas respaldan la idea de que la educación puede ser un factor crucial para reducir la exclusión social y promover la movilidad social”(p.2).

Sin embargo, es importante tener en cuenta que la educación no siempre es accesible de manera equitativa para todos, existen barreras socioeconómicas, culturales y estructurales que pueden limitar el acceso y el éxito educativo de ciertos grupos de personas. Por lo tanto, para lograr los beneficios potenciales de la educación, en términos de desarrollo personal y transformación social, es fundamental abordar estas desigualdades y trabajar por sistemas educativos más inclusivos y equitativos. Esto implica no solo garantizar el acceso a la educación, sino también mejorar su calidad y adaptarla a las necesidades cambiantes de las sociedades contemporáneas. En este contexto, Narro et al. (2012) manifiestan que la

educación es uno de los factores que más influye en el avance y progreso de personas y sociedades.

Además de proveer conocimientos, la educación enriquece la cultura, el espíritu, los valores y todo aquello que caracteriza a los seres humanos, la educación impacta el desarrollo económico de los países. En un mundo donde la competencia comercial es fuerte, el peso de la ciencia y la tecnología es cada vez mayor, la educación siempre ha sido relevante para el desarrollo. La educación adquirió mayor importancia en la actualidad por el avance de la ciencia y sus aplicaciones, así como por el desarrollo de los medios, las TIC y por las situaciones sanitarias presentadas.(Barreto y Díaz, 2017)

La pandemia por COVID-19 es en una variable educativa para tener en cuenta, Lonescu et al. (2020) “afirman que la forma de impartir la educación cambió debido al coronavirus” (p.1). Así mismo, Crespo y Palaguachi (2020) indican que hoy en día el hogar es el único espacio posible para realizar esta actividad y evitar los riesgos de un contagio. La educación en estas condiciones solo fue posible gracias a la tecnología que ofrece el mundo actual.

Carpio et al. (2022) manifiestan que la pandemia de COVID-19 afectó los sistemas educativos mundiales, especialmente en países en desarrollo. Por ello, los gobiernos y administradores educativos acogieron medidas de contingencia para mantener el funcionamiento y la sostenibilidad de sus sistemas educativos. Dicho lo anterior, López-Aguilar et al. (2022) declaran que los efectos de la pandemia se sintieron fuertemente en los contextos educativos. En el caso de la educación universitaria surgió un cambio de modelo de enseñanza presencial a uno totalmente virtual. Como se demostró, la tecnología formó parte de la experiencia de aprendizaje en la pandemia, y su presencia será más notoria con el tiempo.

La pandemia del COVID-19 en América Latina obligó a pasar de un modelo educativo presencial a uno a distancia, afectado por emergencias, precariedad tecnológica y falta de planificación. Esto aumentó la necesidad de alfabetización mediática en la región (Mateus

et al., 2022). Ahora bien, si la tecnología forma parte de la enseñanza y aprendizaje es preciso conocer sus impactos y condiciones para usarla. La virtualidad, en mayor o menor medida, formará parte de una nueva normalidad, y todas las actividades humanas deben adaptar sus procesos más tradicionales a entornos virtuales.

La educación es una de las primeras actividades que cambió con la pandemia del COVID-19. De acuerdo con García et al. (2022) la crisis sanitaria provocada por el COVID-19 generó que las instituciones educativas de todos los niveles se trasladen a medios y plataformas digitales, implementen estrategias metodológicas para que los estudiantes continúen con sus clases y eviten retrasos en el plan curricular.

Según Villapudua et al. (2021) la precariedad digital, falta de concentración, ruido, tareas del hogar, extrañar a los amigos y profesores, falta de tiempo, organización y trabajos en equipo son algunas de las dificultades presentes en el proceso de aprendizaje. En la virtualidad es importante crear estrategias de sensibilidad docente, rediseños curriculares y comités de apoyo socioemocional a los estudiantes en crisis, para lograr una adaptación adecuada a la nueva normalidad y que el impacto no sea fuerte.

Cabero et al. (2020) manifiestan que las instituciones universitarias necesitan un cambio radical de la digitalización, reflexionan sobre los azotes que el COVID-19 produjo a la educación superior, también analizan el imaginario social de la formación a distancia virtual, las desigualdades y brechas digitales en el acceso a las tecnologías o la falta de formación y desarrollo de competencias digitales del docente y discente ante la situación sobrevenida. En el contexto de la formación profesional universitaria, Colina (2007) “define a las IES como espacios sociales con características temporales afectadas por circunstancias culturales, dándoles un carácter distintivo” (p.4).

Las IES cumplen funciones y actividades comunes: enseñanza, diseño de programas académicos, investigación, becas, dotación de personal, infraestructura, ambiente académico, desarrollo y difusión del conocimiento con programas de investigación. En

perspectiva general, son características que las unifican en su praxis (Bartolomé y Grané, 2013).

Alarcón et al., (2014) manifiestan que la innovación de la calidad de enseñanza y aprendizaje en las IES es primordial. Ahora bien, aunque es difícil transformar una institución, sin cambios no avanza el conocimiento. En consecuencia, “surge la necesidad de visualizar nuevas competencias digitales para integrar el conocimiento tecnológico de los medios con el conocimiento didáctico de su uso” (Vidal et al., 2021,p.139). Por esta razón, es importante investigar y conocer sobre las competencias digitales.

En el ámbito de la educación superior, el mundo académico muestra entusiasmo por el desarrollo de las capacidades relacionadas con las competencias digitales (Cabero-Almenara et al., 2022). Así como por el incremento de las habilidades y actitudes que permitan un ecosistema digital completo (Suárez-Guerrero et al., 2021).

Sánchez (2021) afirma que la educación superior en los campos de la Ingeniería Informática muestra un entusiasmo por el desarrollo de diversas competencias relacionadas con las tecnologías de ello surge la necesidad del adecuado uso de herramientas complementarias como las herramientas de CN, enfocadas a mejorar las destrezas de programación alineadas a los estándares y al contexto de la CA.

El desarrollo del talento de los ingenieros en informática es esencial para garantizar la calidad, la innovación y la competitividad en el campo de la tecnología. La formación académica en ingeniería en informática juega un papel crucial en el desarrollo de las habilidades y conocimientos necesarios para tener éxito en este campo en constante evolución.

Vera y Argüello (2019) manifiestan que el aprendizaje de lenguajes de programación constituye una de las necesidades a atender por todo centro educativo, en cualquier nivel de enseñanza. La demanda de modelos novedosos de formación, acordes al contexto cultural y tecnológico de la época, les permite disponer de una visión facilitadora, de un avance

adecuado y pertinente (p.40). La integración de la tecnología al proceso de enseñanza-aprendizaje, además de un reto, resulta fundamental en el desarrollo de esos modelos (Cabero et al., 2015).

El surgimiento de nuevos paradigmas educativos toma importancia y se relaciona con la integración curricular de la tecnología en la educación, creando en el docente competencias distintas a las tradicionales; en este sentido Del Prado y Lamas (2014) comentan que “La enseñanza de programación trata de enseñar a operar lenguajes y no a programar, porque programar implica primero diseñar lógicamente la solución y después codificar usando un lenguaje” (p. 111).

Balladares et al. (2016) “manifiestan que la tecnología permite crear ambientes personales de aprendizaje y en red, que estimulan y dinamizan el estudio, tanto individual como grupal, especialmente en el área de la informática”(p.148).

En la literatura se presentan artículos del uso de la CN y su enfoque en el contexto educativo (Moro-Cabero y Llanes-Padrón, 2019). Los modelos de gestión y almacenamiento basado en la nube aplicados en la educación (Askhoj et al.,2011). Habilidades computacionales soportadas por CN (Benson et al., 2011), pero no se combinan con la CA como parte de un modelo pedagógico; de ello la importancia de la propuesta generada del uso de este tipo de herramientas de CN y principios de la CA.

La educación superior no es igual en todo el mundo, por ejemplo, la ingeniería informática ha evolucionado de acuerdo con la realidad de cada país y de las necesidades del entorno. En el caso de Ecuador, la enseñanza de la programación se centra en el empleo de plataformas, aplicaciones y en la creación de contenidos, pero no en la investigación de nuevos modelos educativos flexibles, en ambientes de aprendizaje, que generen una adquisición de conocimiento a partir de la interacción; por ello en esta investigación se propone el empleo de la CN y CA como herramientas de apoyo a la educación superior y su

importancia para el desarrollo de las habilidades de programación en los estudiantes de ingeniería informática.

Cooperberg (2002) expresa que no todas las TIC son iguales, además de las herramientas propias para acompañar la docencia como Microsoft, PowerPoint o aulas virtuales, hay tecnologías de información que generan nuevos lenguajes y modos de representación, y facilitan la creación de nuevos escenarios de aprendizaje. Por ello, las IES no pueden permanecer al margen, ya que deben conocer y utilizar estos nuevos lenguajes y formas de comunicación, para mejorar el proceso de enseñanza y aprendizaje (p.9).

De acuerdo con Severin (2013) las TIC se deben emplear de manera apropiada, para desarrollar sociedades más democráticas e inclusivas en las que se fortalezcan la colaboración, creatividad y distribución justa del conocimiento científico y se contribuya a una educación equitativa y de calidad para todos (p.17). Esto supone el desarrollo de una mirada pedagógica sobre las herramientas de CN y la CA que, sumada al desarrollo tecnológico, de lugar a más espacios de aprendizaje que permitan superar limitaciones de gestión del tiempo y presencialidad física, para ello la educación usa diversas plataformas que facilitan el proceso de enseñanza – aprendizaje (De Pablos Pons et al., 2011,p.37). A nivel universitario, el estudiante dispone de diferentes herramientas de carácter colaborativo basadas en las TIC que facilitan el intercambio de información (Domínguez et al., 2011).

Todo el desarrollo tecnológico debe ser atendidos por una mirada pedagógica (Suárez-Guerrero et al., 2024) exigente y esta es la motivación de la tesis. Los modelos pedagógicos son marcos de representación que integran muchas variables para favorecer el aprendizaje, entre ellas la tecnología (Merino y Lizandra, 2021).

Gumuslouglu y Ilsev (2009) expresa “Es ahí donde cabe la innovación y no solo en la tecnología, es decir, la innovación no llega por usar tecnología en educación, sino por pensarla y diseñarla en un marco pedagógico que le da sentido” (p.10).

Cabe mencionar, además, que dentro de la innovación educativa “la CN y CA se han convertido en aspectos elementales y, más aún, en el entorno de aprendizaje y desarrollo de habilidades en los profesionales de ingeniería, en especial del área de la informática” (Quezada y Suárez-Guerrero, 2021, p.11). El interés de la tesis es probar un desarrollo educativo con esas herramientas. Se debe agregar que la CN se está convirtiendo en uno de los mejores recursos para aportar soluciones que satisfagan la necesidad actual de una mayor colaboración entre empresas, educación y sociedad (Sclater,2021).

El aprendizaje de programación es importante durante la vida profesional del ingeniero en Informática, ya que conlleva transformar tareas mecánicas en tareas automatizadas, de igual forma le permitirá gestionar procesos enfocados en alinear los servicios de Tecnologías de la Información en adelante (TI) a las necesidades de las empresas (Quezada-Sarmiento y Mengual-Andrés, 2019,p.4). Por esta razón, la educación usa diversas plataformas para el proceso de enseñanza – aprendizaje (Kölling y Rosenberg, 2001).

Hay que mencionar que uno de los objetivos de la educación es desarrollar habilidades y potenciarlas con el proceso pedagógico desde una perspectiva científica y tecnológica, pretendiendo racionalizar los modelos de enseñanza, proponiendo como alternativa la descripción de los aprendizajes. De la misma manera, la articulación de un conjunto de conocimientos es un paso esencial hacia el desarrollo de una profesión, puesto que representa un amplio consenso respecto a lo que un profesional de la ingeniería informática debe conocer, en especial respecto a las destrezas computacionales de programación (Martínez et al., 2018,p.22).

Resende (2017) manifiestan las nuevas tecnologías y paradigmas computacionales pueden ser empleados en diversos campos, entre ellos el educativo. Sirven para el diseño, búsqueda, presentación, intercambio y reutilización de material, debido a que la tecnología permite almacenar, organizar, replicar, difundir, transformar y ser accesible, lo que conlleva el ahorro de tiempo y recursos, no obstante, la tecnología no opera sola, debe integrar un modelo pedagógico (p.98).

Kiili y Ketamo (2018) manifiesta que la necesidad lleva a pensar con seriedad el adecuado uso de herramientas complementarias como las herramientas de CN, enfocadas a la mejora de destrezas de programación, que a su vez estén alineadas a los estándares y al contexto de la CA (p.258). Este ámbito de la CA se interesa en entornos educativos (Lucas et al., 2021). Ambas variables del mundo tecnológico pueden ser significativas para el aprendizaje en la educación superior (Torres et al., 2022).

Generalizando, esta tesis doctoral se enfocó en el Diseño de un Modelo Pedagógico basado en CN y CA para el desarrollo de las habilidades de programación, mediante prácticas e interacciones en las que el estudiante desarrolló habilidades que dieron lugar a una nueva perspectiva sobre su aplicación en la resolución de problemas de aprendizaje, de codificación y paradigmas computacionales. Asimismo, se buscó comprobar si el uso de dichas herramientas hacía posible generar una mayor interacción, fomentando el trabajo en equipo entre los profesionales en formación, lo que dio lugar a una retroalimentación positiva entre docentes y estudiantes. La tesis, se apoya en las siguientes ideas:

- Enfoque basado en el desarrollo de habilidades de programación en la ingeniería informática: La necesidad de desarrollar habilidades de programación en los estudiantes de ingeniería en informática es crucial en el mundo actual, donde la tecnología desempeña un papel cada vez más importante en todos los aspectos de la vida. El diseño de un modelo pedagógico centrado en estas habilidades es relevante y puede tener un impacto significativo en la formación de los estudiantes de informática, esto fue validado con los casos de estudio desarrollados en el marco de la presente tesis doctoral.
- Uso de herramientas de CN y CA: La inclusión de CN y CA en el modelo pedagógico proporcionó un enfoque innovador y centrado en el estudiante. Estas herramientas los ayudaron a desarrollar habilidades de pensamiento crítico, resolución de problemas y trabajo en equipo, que son fundamentales en el campo de la informática, especialmente en la vida profesional.

- Prácticas e interacciones: La incorporación de prácticas e interacciones en el modelo pedagógico es fundamental para proporcionar a los estudiantes experiencias y oportunidades de aprendizaje activo. Estas actividades pueden ayudarlos a aplicar los conceptos teóricos en situaciones reales y a desarrollar una comprensión más profunda de los temas tratados.
- Fomento del trabajo en equipo: El énfasis en el fomento del trabajo en equipo entre los estudiantes es clave para prepararlos para el entorno laboral actual, donde la colaboración y la comunicación son esenciales. El modelo pedagógico proporcionó oportunidades para que los estudiantes trabajen juntos en proyectos y actividades colaborativas, lo que les permite desarrollar habilidades de trabajo en equipo y liderazgo.
- Fomento del trabajo en equipo: La retroalimentación entre docentes y estudiantes fue esencial para el éxito del modelo pedagógico, esto dio lugar a una retroalimentación constructiva y oportuna que ayudó a los estudiantes a mejorar su desempeño y a desarrollar una comprensión más profunda de los conceptos. Además, la retroalimentación positiva aumentó la motivación y el compromiso de los estudiantes con el proceso de aprendizaje. En resumen, la propuesta de tesis doctoral aborda temas importantes para la educación en informática y tiene el potencial de contribuir significativamente al desarrollo de los estudiantes en este campo. Esta propuesta de modelo pedagógico basado en la CN y CA forma parte de los paradigmas computacionales basados en las estrategias metodológicas necesarias para la enseñanza de programación en el área de la informática. Esta tesis, por tanto, busca articular el desarrollo informático dentro de la mirada pedagógica.

1.2 Importancia del tema

Entender la CN y la CA desde la perspectiva de los modelos pedagógicos requiere analizar cómo estas tecnologías se integran y complementan los procesos de enseñanza y aprendizaje. A continuación, se presenta una explicación de cada tecnología y su relación con los modelos pedagógicos:

La CN se refiere al uso de recursos informáticos (servidores, almacenamiento, bases de datos, redes, software, análisis e inteligencia) a través de Internet para ofrecer mayor flexibilidad, escalabilidad y eficiencia en el procesamiento de datos (Akamai Technologies, 2024). Desde una perspectiva pedagógica, la CN ofrece varias ventajas:

- **Accesibilidad:** Permite el acceso a materiales educativos, software y plataformas desde cualquier lugar y momento, fomentando el aprendizaje autónomo y a distancia.
- **Colaboración:** Facilita el trabajo colaborativo a través de plataformas que pueden ser utilizadas simultáneamente por varios usuarios, promoviendo metodologías activas como el aprendizaje basado en proyectos.
- **Escalabilidad:** Se adapta fácilmente a las necesidades cambiantes de las instituciones educativas, permitiendo ampliar o reducir recursos según sea necesario.
- **Innovación:** Ofrece la posibilidad de incorporar rápidamente nuevas herramientas y tecnologías emergentes en el proceso educativo.

Gracia & Fabila, 2011 “expresan que la integración de la CN en los modelos pedagógicos puede facilitar enfoques más personalizados y adaptativos del aprendizaje, donde los recursos se ajustan a las necesidades individuales de los estudiantes” (p.15).

Acerca del término Affective Computing en español “Computación Afectiva” (CA), Picard (1997) lo definió “como el área que estudia cómo los sistemas computacionales pueden reconocer y reaccionar a las emociones humanas, expresándolas por medio de una interfaz

computacional”(p.20). Este campo de estudio considera dos características importantes en relación con las emociones humanas: la capacidad de reconocer el estado emocional de otro individuo y la experiencia social en la que el individuo tiene la sensación de que su estado emocional es entendido por otras personas (Picard y Klein, 2002,p.150).

Ma y Yarosh (2021) plantean que la CA es un campo que cierra la brecha entre el afecto humano y la tecnología computacional, siendo un avance técnico notable. En la CA, las máquinas deben detectar y reconocer las emociones humanas y adaptar su comportamiento a ellas (Puviani et al., 2021).

Tuncay (2010) expresa:

Que la CA es un campo naciente situado en la intersección de la inteligencia artificial con las ciencias sociales y del comportamiento. Estudia cómo se perciben y expresan las emociones humanas, lo que permite diseñar agentes y sistemas inteligentes que imitan el comportamiento y mejora la comunicación de manera efectiva entre colaboradores humanos. La investigación de la CA ha visto avances significativos recientemente y está realizando una transformación crítica de estudios exploratorios a aplicaciones en el área de investigación emergente conocida como CA aplicada (p.939).

“ En la última década, aumentó significativamente el interés por los comportamientos emocionales humanos, lo que influye también en el futuro de la computación multimodal interactiva ” (Moghimi et al., 2020, p.52).

La CA se centra en el desarrollo de sistemas y dispositivos capaces de reconocer, interpretar, procesar y simular las emociones humanas.

En el contexto educativo, esta tecnología permitió:

- Personalización: Se adaptaron los procesos de aprendizaje en función del estado emocional y cognitivo del estudiante, mejorando su motivación.
- Retroalimentación emocional: Proporcionó la retroalimentación a educadores sobre las respuestas emocionales de los estudiantes hacia el material o las actividades

propuestas, permitiendo ajustes en tiempo real para mejorar la experiencia de aprendizaje.

- Desarrollo socioemocional: Apoyo al desarrollo de habilidades socioemocionales a través de programas y juegos que enseñan empatía, reconocimiento de emociones y regulación emocional.

Aranha et al., 2021 “manifiesta que uno de los enfoques que pueden promover la participación del usuario es la CA, basada en la premisa de reconocer el estado emocional del usuario y ajustar la aplicación informática para responder a dicho estado en tiempo real “ (p.885). Así mismo, Salcedo et al.,(2021) expresan que las emociones desempeñan un papel muy importante en la adquisición, procesamiento de información y en el aprendizaje de las personas; en consecuencia, dentro de la inteligencia artificial, en los últimos años se ha visto un crecimiento vinculado a las investigaciones sobre el reconocimiento de las emociones como parte del área de la CA que aportan en el campo educativo (p.4).

La aplicación de la CA en los modelos pedagógicos implica un enfoque más holístico de la educación, donde no solo se atienden los aspectos cognitivos del aprendizaje, sino también los emocionales y sociales. La CN es una tecnología que puede proporcionar muchos beneficios en el campo de la educación. La utilización de la tecnología de computación en la nube es una necesidad en el entorno universitario, donde el aprendizaje colaborativo es una característica principal.

La incorporación de la CN y la CA en los modelos pedagógicos sugiere un cambio hacia enfoques más flexibles, personalizados y centrados en el estudiante. Esto puede verse reflejado en modelos como el aprendizaje invertido, el aprendizaje basado en proyectos, el aprendizaje adaptativo y el aprendizaje socioemocional, donde la tecnología actúa como facilitadora para satisfacer las necesidades individuales de aprendizaje y promover un ambiente educativo más inclusivo y participativo.

La clave para una integración exitosa de estas tecnologías radica en la formación y capacitación de los docentes, la infraestructura tecnológica adecuada y un enfoque pedagógico que propicie el desarrollo integral del estudiante.

Entender la CN y la CA desde la perspectiva de los modelos pedagógicos implicó explorar cómo diferentes enfoques de enseñanza pueden facilitar el aprendizaje de estas tecnologías avanzadas. Ambos campos son interdisciplinarios y emergentes, lo que requiere estrategias pedagógicas adaptativas y reflexivas para su enseñanza efectiva. El panorama no es sencillo, como tomar una herramienta, sino que hay que explorar en la pertinencia de enfoques de aprendizaje como:

1. Enfoque conductista: Desde esta perspectiva, el aprendizaje de la CN y CA podría estructurarse en torno a la repetición y la práctica dirigida. Por ejemplo, los estudiantes aprendieron a través de la repetición de tareas como configurar y administrar entornos de nube o programar sistemas que reconocen emociones, con énfasis en la obtención de respuestas correctas a través de la retroalimentación inmediata.
2. Enfoque cognitivo: Este modelo enfatiza el entendimiento profundo y el procesamiento de la información. En la enseñanza de la CN, se incluyó el estudio detallado de arquitecturas de nube y modelos de despliegue. Para la CA, los estudiantes podrían explorar los fundamentos de la psicología emocional y las técnicas de inteligencia artificial que permiten a las máquinas interpretar las emociones humanas. Los cursos podrían incluir tareas que fomenten la reflexión crítica sobre cómo estas tecnologías impactan en la sociedad.
3. Enfoque constructivista: Aquí, los estudiantes construyen activamente su propio conocimiento. En el contexto de estas tecnologías, esto podría dar lugar a proyectos donde los estudiantes desarrollen sus propias aplicaciones basadas en la nube o creen prototipos de sistemas afectivos. Este enfoque promueve la exploración

personal y el aprendizaje a través del hacer, lo que puede ser especialmente efectivo para comprender conceptos tecnológicos complejos y su aplicación práctica.

4. Enfoque sociocultural: Este modelo resaltó la importancia de la interacción social en el aprendizaje. Podría aplicarse organizando debates en clase sobre los aspectos éticos y sociales de la CA y la CN o proyectos grupales donde los estudiantes trabajen en equipo para desarrollar soluciones basadas en estas tecnologías, aprendiendo unos de otros, compartiendo conocimientos y habilidades de programación.
5. Enfoque de aprendizaje basado en problemas (ABP): Este enfoque es particularmente relevante para tecnologías emergentes como estas. Los estudiantes enfrentaron problemas reales en los que usaron la CN y la CA para resolverlos. Por ejemplo, diseñaron un sistema en la nube para gestionar grandes volúmenes de datos o desarrollar una aplicación que utilice reconocimiento emocional para mejorar la experiencia del usuario en situaciones del mundo real.

Cada uno de estos enfoques tiene el potencial de enriquecer el aprendizaje de la CN y CA, haciéndolo más relevante y aplicable a situaciones reales y fomentando una comprensión más profunda de sus capacidades y limitaciones.

Sangrà et al., (2020) manifiestan:

Que el uso de la tecnología de la CN y CA en educación superior no es un tema solo de herramientas sino de competencias y habilidades, que abre una línea de investigación emergente, especialmente en la educación digital y la actual educación digital de emergencia (p.13).

En este marco, la CN aporta un enfoque que se integra a otros elementos necesarios en el aprendizaje y permite explotar la colaboración en red. Formativamente hablando, la CN en el aprendizaje de la programación aporta al alumnado la posibilidad de transformar tareas comunes en tareas automatizadas, gestionar procesos enfocados en alinear los servicios tecnológicos a las necesidades de las empresas o servicios, así como la posibilidad de añadir

el factor de Investigación +Desarrollo +innovación (I+D+i) para crear un producto innovador basado en el empleo de CN (Quezada-Sarmiento et al., 2018,p.4).

Romero et al., (2020) dentro de este contexto expresa:

El uso de CN permitió generar un ambiente de mayor interactividad entre docentes y estudiantes, generando redes de colaboración a través de las herramientas de la nube que permiten una mayor interacción entre los estudiantes, los cuales pueden generar un entorno social colaborativo para el desarrollo de las habilidades de programación. Se debe agregar que la CN es una tecnología que proporcionó muchos beneficios en el campo de la educación. La utilización de la tecnología de CN es una necesidad en el entorno universitario, donde el aprendizaje colaborativo es una característica principal (p.5).

Según Alim y Hai (2017), la CN es importante porque puede proporcionar todos los recursos necesarios para el aprendizaje electrónico, en particular los servicios más intensivos en informática, como mundos virtuales, simulaciones y transmisión de video. Dentro del mismo contexto, en la educación convencional el estudiante tiene que adaptarse a las condiciones que impone la escuela, al profesor, incluso a sus compañeros; pero si realmente queremos que el aprendizaje sea efectivo, deberemos tener en cuenta las capacidades y la disposición al aprendizaje de cada individuo, y adaptar la manera de enseñar al ritmo de cada uno. Hoy en día esto es posible gracias a la tecnología y más aún con el soporte de herramientas como las de CN y CA.

En cuanto al modelo pedagógico propuesto, se presenta como una herramienta conceptual para entender mejor el contexto de la enseñanza de la programación por medio de una representación del conjunto de relaciones que describen el proceso respectivo que es soportado con el uso de las CN y CA.

En el modelo pedagógico desarrollado se puede entender la representación abstracta-conceptual de las relaciones que predominan en el fenómeno de la enseñanza, además, acopia aspectos de relevancia para analizar, describir, explicar, simular, controlar, predecir y explorar los fenómenos, y procesos educativos que se enfocan en la mejora de las

habilidades blandas y técnicas de la programación. La conceptualización del modelo pedagógico propuesto obedece a una coherencia en el problema esencial de cada etapa de la formación en el área de la ingeniería informática. Es importante mencionar que dicha formación depende del tipo de estudiante que se desea formar y a su vez de la sociedad que se desea construir.

De acuerdo con Cantor et al. (2019) "pensar en la relación de las teorías pedagógicas con la concepción del hombre y la sociedad, con los intereses y necesidades de los profesores que evidencian el modo de actuación profesional" (p.104).

1.3 Contexto de la investigación

Es importante considerar que el área educativa se ha visto en la necesidad de crear espacios sin limitantes de tiempo ni capacidad de almacenamiento, es así como la educación utiliza diversas plataformas para el proceso de enseñanza – aprendizaje (Jan et al.,2011).

El empleo de diversas herramientas tecnológicas en educación superior integra competencias y habilidades, además, une a docentes, investigadores y estudiantes que generan conocimiento. En la actualidad se requieren modelos educativos abiertos y flexibles al servicio de los estudiantes, que potencien el desarrollo de competencias aplicables a nivel profesional y se vinculen con las diferentes áreas de conocimiento, en especial en la Ingeniería Informática.

La presente propuesta de tesis doctoral se enfocó en el uso de CN y CA que, mediante las prácticas e interacciones, promovió que los estudiantes de la Universidad Técnica Particular de Loja (UTPL), Universidad Nacional de Loja (UNL) y Universidad Internacional del Ecuador (UIDE) que cursaron asignaturas relacionadas con la programación- desarrollasen habilidades que generen una visión sobre su aplicación a la resolución de problemas de aprendizaje de codificación y paradigmas computacionales. Así mismo, el empleo de dichas herramientas permitió generar una mayor interacción, fomentando el trabajo en equipo entre los profesionales en formación y desarrolló una retroalimentación de

los temas abordados entre docentes y estudiantes.

La tesis doctoral planteada desarrolló un estudio de la CN y CA en el proceso de aprendizaje, el cual será la base para la propuesta de un modelo de enseñanza de paradigmas computacionales basados en estrategias metodológicas para la enseñanza de programación en el contexto del área de la informática.

1.3.1 Preguntas de investigación de la tesis doctoral

La tesis busca dar respuesta a la siguiente pregunta central:

- ¿Cómo articular la CN y la CA en un modelo de aprendizaje que favorezca el aprendizaje de la programación en el contexto de ingeniería informática?

Para ello, se trabaja en una serie de preguntas adicionales:

- ¿Qué información académica existe en las bases de datos científicas referente a modelos pedagógicos basados en CA y CN?
- ¿Cómo contribuye el uso de herramientas de CN y la CA al desarrollo de las destrezas de programación en el contexto de ingeniería informática?
- ¿Cuál es la importancia de la CA en la resolución de problemas de aprendizaje de codificación y paradigmas computacionales en los estudiantes de ingeniería informática?
- ¿Qué procesos de aprendizaje y de enseñanza se estimulan cuando se trabaja con CN y CA en el aprendizaje de la programación en el contexto de ingeniería informática?

1.4 Objetivos de la tesis doctoral

Para atender las preguntas formuladas anteriormente, este trabajo busca de forma general:

- Diseñar un Modelo Pedagógico basado en herramientas de CN y CA para el desarrollo de las habilidades de programación en el área Informática.

De forma específica:

- Conocer el estado del arte referente al uso de herramientas de CN y la relación de la CA en el desarrollo de las habilidades de programación en el área Informática.

- Analizar el uso de herramientas de CN y la relación de la CA para la mejora de las habilidades de programación en alumnos del área de informática que cursaron asignaturas con componente de programación de las tres universidades ecuatorianas de los casos de estudio desarrollados.

- Valorar y establecer las estrategias metodológicas para la enseñanza de programación en el área de informática en el ámbito de la educación superior.

- Articular la CN y la CA en un modelo de aprendizaje que favorezca el aprendizaje de la programación en el contexto de ingeniería informática.

1.5 Metodología de la investigación

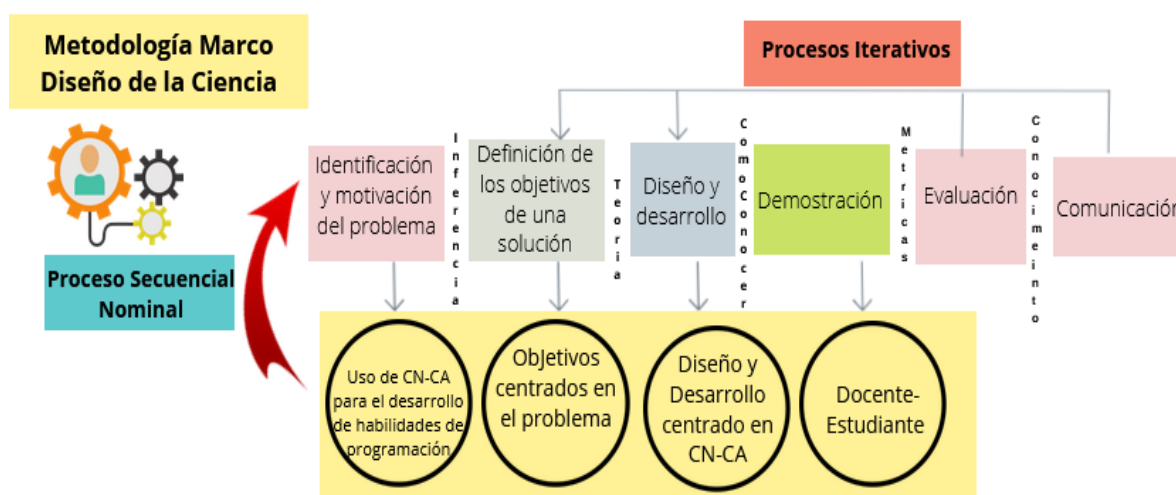
La presente tesis doctoral se desarrolló bajo la metodología de investigación basada en Diseño de la Ciencia, en adelante (DC), propuesta por Hevner et al. (2003) y complementada con la revisión sistemática de la literatura (Kitchenham y Charters, 2007). DC fue utilizada para desarrollar y estudiar tecnologías que se fundamentan en la aplicación práctica y en la creación de artefactos. Hevner y sus colegas establecen una serie de guías y principios que son esenciales para llevar a cabo investigaciones efectivas en este campo. La metodología se puede describir a través de varios componentes clave:

1. Identificación y motivación del problema de investigación: Se identificó un problema relevante en un contexto real que requiere una solución tecnológica, en nuestro contexto, proponer el desarrollo de un modelo pedagógico basado en CA y CN que aporte al desarrollo de las habilidades de programación.
2. Objetivos de la solución: Se especificaron los objetivos que la solución de diseño debe alcanzar, los cuales suelen ser innovadores y útiles. Dichos objetivos se presentan en la sección 1.4 de la presente tesis doctoral.
3. Diseño y desarrollo: Se crearon diferentes artefactos para resolver el problema propuesto. Estos artefactos fueron modelos, métodos, procesos y producto.
4. Demostración: Se demostró que los artefactos resolvieron el problema en uno o más escenarios. Esto se realizó a través de ejemplos, experimentos, simulaciones, etc.
5. Evaluación: Se evaluó el artefacto para determinar cuán bien resuelve el problema. Esto implica la medición de cómo el artefacto satisface los objetivos de la solución a través de métodos como pruebas de campo, estudios de caso, y análisis de datos.
6. Comunicación: Los resultados del diseño y las evaluaciones se comunicaron a las audiencias relevantes. Esto incluye publicaciones académicas para investigadores y presentaciones, o demostraciones, a otros stakeholders como practicantes o usuarios

finales. En la sección 1.6 se presentan los resultados de las comunicaciones científicas generadas en la presente tesis doctoral.

El ciclo del DC fue interactivo, lo que permitió mejoras continuas del artefacto basadas en los comentarios y resultados de las evaluaciones. Esta metodología no solo enfatizó la creación de soluciones tecnológicas, sino también la generación de conocimiento teórico que puede guiar futuras investigaciones y prácticas. En la Figura 1 se pueden observar las fases antes mencionadas. Se debe agregar que los detalles, de cada una de las fases del DC, serán descritas en el capítulo 5 de la presente tesis doctoral.

Figura 1
Fase de diseño de la ciencia



Nota. La Figura 1 presenta las fases de la metodología de DC.

Las razones por las que se seleccionó el DC en la tesis doctoral son las siguientes

El DC aplicado en la presente tesis doctoral fue un paradigma de investigación en el que el investigador responde a preguntas pertinentes a los problemas humanos a través de la creación de artefactos innovadores, contribuyendo con nuevos conocimientos al cuerpo de la evidencia científica. En nuestro caso, se centran en el uso de CN y CA en el proceso de desarrollo de destrezas tecnológicas, enfocadas a la resolución de problemas de aprendizaje de codificación y paradigmas computacionales en profesionales en formación de Ingeniería Informática.

El principio fundamental del DC es que el conocimiento y la comprensión de un problema

de diseño y su solución se adquieren en la construcción y aplicación de un artefacto, en nuestro caso enfocado al campo educativo, en especial en la adquisición de destrezas de programación.

Además, como parte del proceso metodológico de la tesis, se realizó una revisión sistemática de la literatura que ha sido propuesta en la Ingeniería de Software por Kitchenham y Charters (2007) como un método para presentar conclusiones confiables sobre un área de investigación que recoge sistemáticamente las pruebas de calidad, en este caso en la enseñanza de paradigmas computacionales soportados con el uso de herramientas de la CN y CA.

Se debe agregar que la razón por la que se realizó la revisión sistemática de literatura en la tesis doctoral se fundamenta en que esta metodología es una estrategia de búsqueda formal. Además, facilita de una manera más fiable la localización de las contribuciones científicas que podrían proporcionar respuestas pertinentes a los contextos de investigación. El seguir una estrategia de búsqueda formal permitió las prácticas recomendables en contextos de investigación como la capacidad de repetición y los exámenes externos de esta contribución (Quezada et al., 2020, p.5).

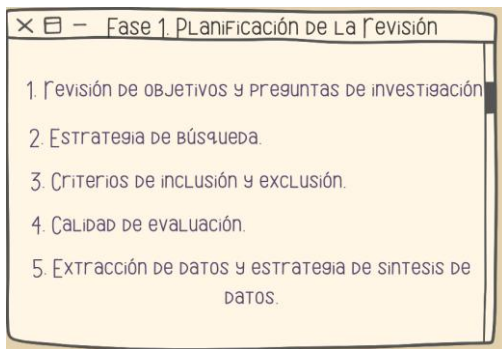
La revisión sistemática de literatura se ejecutó en cada una de las fases, de acuerdo con el protocolo establecido (Kitchenham y Charters, 2007). En el desarrollo del protocolo se especificó: (i) el objetivo y las preguntas de investigación, (ii) la estrategia de búsqueda, (iii) los criterios explícitos de inclusión y exclusión, (iv) los criterios para evaluar cada estudio, y (v) la estrategia para sintetizar los datos extraídos y presentar la evidencia científica que soporte la investigación.

Las fases desarrolladas fueron:

- Planificación. - Se desarrolló el protocolo de revisión en donde se definen los métodos para realizar la revisión sistemática de la literatura.

En la Figura 2 se puede apreciar las etapas de la planificación de la revisión.

Figura 2
Planificación de la Revisión

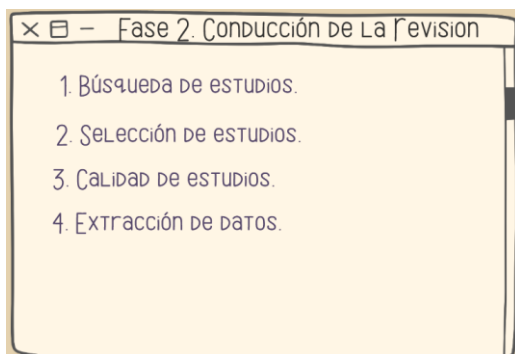


Nota. Etapas de la planificación de la Revisión Sistemática de la Literatura aplicada. Adaptado de (Kitchenham y Charters, 2007).

- Conducción. - Ejecución del protocolo propuesto.

En la Figura 3 se puede apreciar las etapas de conducción de la revisión.

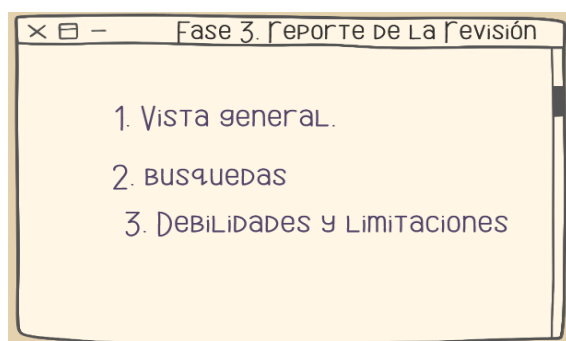
Figura 3
Conducción de la Revisión



Nota. Etapas de la conducción de la revisión sistemática de la literatura aplicada. Adaptado de (Kitchenham y Charters, 2007).

- Reporte. - Se brindó a la comunidad científica los resultados generados de la ejecución y la revisión sistemática de la literatura. En la Figura 4 se puede visualizar las fases del reporte de la revisión.

Figura 4
Reporte de la Revisión



Nota. Etapas de reporte de la Revisión Sistemática de la Literatura aplicada. Adaptado de (Kitchenham y Charters, 2007).

En general, la metodología de la tesis se puede apreciar de forma detallada en el capítulo 3 donde, además del protocolo de desarrollo mediante la revisión sistemática de la literatura propuesta en la presente tesis doctoral, se describen las fases del DC aplicado en la parte empírica en la que se desarrollaron casos de estudios con el uso de CN y principios de la CA. Se ha buscado en lo posible conocer toda la información generada al respecto para partir de allí, y someterla a validación en base a la itinerancia constante del modelo gracias a la DC.

El lugar de desarrollo de los casos cuasi experimentales fue Ecuador, provincia de Loja, cantón Loja, ciudad Loja, en tres instituciones de educación superior (IES). Las IES fueron la Universidad Técnica Particular de Loja (UTPL), la Universidad Internacional del Ecuador (UIDE) y la Universidad Nacional de Loja (UNL).

Las características más importantes de los seis casos cuasiexperimentales se detallan a continuación:

1. En el caso cuasi experimental desarrollado en la UTPL se trabajó con estudiantes que cursaron la asignatura de Programación Avanzada ubicada en el cuarto ciclo de la carrera de Sistemas Informáticos y Computación, con una carga horaria de 4 créditos, considerando que cada crédito equivale a 32 horas académicas. El curso con el cual se desarrolló el caso empírico estaba compuesto por 25 estudiantes, quienes emplearon y desarrollaron las diferentes actividades

académicas relacionadas con la programación soportadas por las herramientas de CN y principios de la CA, para las construcciones de los prototipos propuestos en el marco del aprendizaje de la asignatura. Dentro del desarrollo del proyecto se usó Diseño de la Ciencia para determinar las actividades y entregables a desarrollar, así como metodologías ágiles.

Entre las herramientas *cloud* utilizadas tenemos: *WordPress* con la integración de *plugins* de desarrollo y módulos de acuerdo con los requerimientos del proyecto, así mismo se empleó *Egroupware*, *REDMINE* y *Open Atrium* tanto para gestión del proyecto, como para el código generado. Es importante manifestar que también se utilizó *GitHub* como repositorio y gestor de retroalimentación del código generado en el proyecto.

1. Así mismo, en la UTPL se desarrolló un segundo caso cuasi experimental con los estudiantes que cursaron la asignatura de Programación Web, con una carga horaria de 128 horas académicas correspondientes al sexto ciclo. El curso con el cual se desarrolló el caso cuasi experimental estaba compuesto por 25 estudiantes, que emplearon y desarrollaron las diferentes actividades académicas relacionadas con la programación de aplicaciones en contextos reales, que fueron soportadas por las herramientas de computación en la nube y principios de la CA para las construcciones de los prototipos propuestos en el marco del aprendizaje de la asignatura.

Así mismo se integró la metodología Mobile-D. Entre las herramientas usadas tenemos Spring Bot, Visual Code para el desarrollo de la aplicación; así como el Tablero Kanban de Tobeio que es una herramienta ágil de gestión de proyectos diseñada para ayudar a visualizar el trabajo, limitar el trabajo en curso y maximizar la eficiencia.

2. En el caso cuasi experimental tres, se trabajó con estudiantes de la UNL que cursaron la asignatura de Proyectos Tecnológicos I (Enfoques de

Programación) ubicada en el séptimo ciclo de la carrera de Computación, con una carga horaria de 120 horas académicas.

El curso con el cual se desarrolló el caso cuasi experimental estaba compuesto por 20 estudiantes, ellos utilizaron y desarrollaron las diferentes actividades académicas relacionadas con la programación soportadas por las herramientas de CN y principios de la CA para las construcciones de los prototipos propuestos en el marco del aprendizaje de la asignatura. El proyecto desarrollado consistió en la creación de una tienda de virtual, soportada con la metodología *Mobile D*. Dentro del mismo caso empírico se desarrolló un prototipo para valorar la parte afectiva del comprador, el cual se ejecutaba luego de la adquisición para con ello medir el estado de ánimo de este. Las herramientas de computación en la nube utilizadas en el caso empírico fueron: Project Open, Jira, StormBoard.

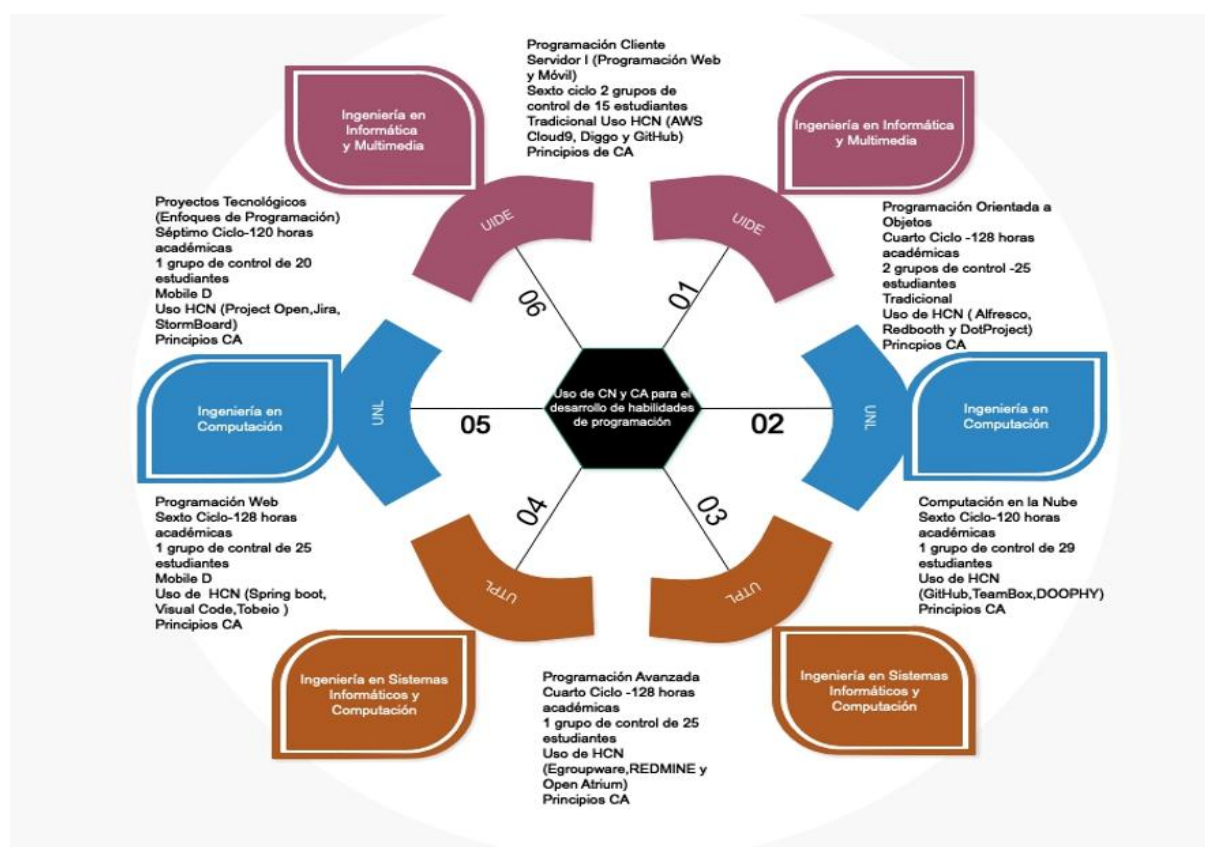
3. En el caso empírico, cuatro se trabajó con estudiantes de la UNL de la carrera de Ingeniería de Computación, en la asignatura de Computación en la Nube del sexto ciclo, de 120 horas académicas. El grupo de control fue de 29 estudiantes. Las herramientas de CN utilizadas fueron GitHub, TeamBOX, DOOOPHY, Principios CA.
4. El quinto caso se desarrolló con los estudiantes de la UIDE que cursan la carrera de Ingeniería en Informática y Multimedia en la asignatura de Programación Orientada a Objetos (POO), correspondiente al Cuarto Ciclo, con una duración de 128 horas académicas con 2 grupos de control de 25 estudiantes.

El grupo uno realizó sus actividades académicas de forma tradicional y el segundo con la utilización de herramientas de CN y CA. Finalmente, en el sexto caso cuasi experimental se trabajó con estudiantes de la UIDE que cursaron la asignatura de Cliente Servidor I (Programación Web y Móvil) con una carga horaria de 128 horas académicas, ubicada en el sexto ciclo de la carrera de Ingeniería Informática y Multimedia.

El curso con el cual se desarrolló el caso cuasi experimental estaba compuesto por dos grupos de control de 15 estudiantes, que utilizaron y desarrollaron las diferentes actividades académicas relacionadas con la programación de aplicaciones en contextos reales, que fueron soportadas por las herramientas de CN y principios de la CA para las construcciones de los prototipos propuestos en el marco del aprendizaje de la asignatura. Dentro del mismo orden de ideas, en el presente caso, se utilizaron las herramientas de computación en la nube: AWS Cloud9, Diggo y GitHub. En los casos cuasi experimentales se realizó la adaptación de los syllabus respectivos para el empleo de las herramientas de CN, así como la introducción de principios de la CA para establecer la mejora de las habilidades de programación de los estudiantes que formaron parte de los casos experimentales desarrollados.

En la Figura 5 se resumen los 6 casos antes mencionados.

Figura 5
Resumen de casos cuasiexperimentales



Nota. La Figura 5 presenta el resumen de los casos cuasi experimentales desarrollados en la presente tesis doctoral.

Para mayor detalle, también se puede recurrir a la tabla 1 donde se presenta el resumen de los casos empíricos desarrollados.

Tabla 1
Características de los casos cuasi experimentales

IES	Asignaturas	Estudiantes	Horas	Herramientas Utilizadas
Universidad Técnica Particular de Loja (UTPL)	Programación Avanzada	25	128	Egroupware, REDMINE y Open Atrium Git Hub WordPress
	Programación	25	128	Spring boot, Visual Code

	Web			Tablero Kanban de Tobeio
Universidad Nacional de Loja (UNL)	Proyectos Tecnológicos I (Enfoques de Programación)	20	120	Project Open, Jira, StormBoard.
	Computación en la Nube	29	128	GitHub, TeamBOX, DOOPHY
Universidad Internacional del Ecuador (UIDE)	Programación Cliente Servidor (Programación Web y Móvil)	15	128	AWS Cloud9, Diggo y GitHub.
	Programación Orientada a Objetos	25	128	Alfresco, DotProject, Redbooth
Total, de estudiantes		139		

Nota. Características de los casos cuasiexperimentales desarrollados en el contexto de la tesis doctoral.

El aporte de modelo pedagógico propuesto es de carácter educativo, se introduce en el campo de la educación un modelo pedagógico basado en CN y CA, soportado por una exploración empírica como parte del desarrollo profesional en el ingeniero informático. El modelo se evaluó mediante 6 casos cuasi experimentales en IES ecuatorianas; donde se articularon diferentes elementos tecnológicos y pedagógicos para crear el modelo pedagógico educativo. La información generada de cada caso empírico fue recolectada por las plataformas institucionales de las universidades participantes; así mismo se empleó la observación directa, diferentes herramientas de CN que permitieron la construcción de los diversos artefactos para el desarrollo del modelo propuesto en la presente tesis doctoral.

1.6 Contribución de la investigación propuesta

El desarrollo de la presente tesis doctoral ha contribuido en la comunidad científica en los siguientes aspectos:

1. Desarrollo de prototipos computacionales basados en CN y CA.
2. Creación de un Modelo Pedagógico de enseñanza de programación en Educación Superior basado en CN y CA.

Dentro de las contribuciones a nivel de producción científica generadas en la presente tesis doctoral destacan:

1.6.1 Artículos en revistas indexadas

Artículo 1. *Principles of the semantic web and affective computing in a sustainable ecotourism through the development of educational web application*

Autor: Pablo Alejandro Quezada Sarmiento

Aporte en la tesis doctoral: Empleo de herramientas de CN, principios de CA y desarrollo de caso experimental en el contexto de la enseñanza de la Programación Web. El artículo es resultado de la aplicación de la CA en el contexto del desarrollo de habilidades de programación.

Revista: RISTI - Revista Ibérica de Sistemas e Tecnologías de Informacao

Estado: Indexado

Base de Datos de Indexación: Scopus

Cuartil: Q3

Factor de Impacto: 0.22

ISSN: 16469895

Cita: Quezada-Sarmiento, P. A., (2019). Principles of the semantic web and affective computing in a sustainable ecotourism through the development of educational web application. *RISTI - Revista Ibérica de Sistemas e Tecnologías de Informacao*, 212-221.

Link: <https://risti.xyz/issues/ristie17.pdf>

Artículo 2. Cloud computing in the training process in web programming

Autores: Pablo Alejandro Quezada- Sarmiento, Cristóbal Nico Suárez -Guerrero

Aporte en la tesis doctoral: Uso de herramientas de CN en el contexto de la programación web, este resultado se generó del caso cuasi experimental.

Revista RISTI - Revista Ibérica de Sistemas e Tecnologías de Informacao

Estado: Indexado

Base de Datos de Indexación: Scopus

Cuartil: Q4

Factor de Impacto 0.15

ISSN: 16469895

Cita: Quezada-Sarmiento, P. A., & Suárez-guerrero, C. (2021). Cloud computing in the training process in web programming. *RISTI - Revista Ibérica de Sistemas e Tecnologias de Informacao*, 2021(E42), 10-19.

Link: <http://www.risti.xyz/issues/ristie42.pdf>

1.6.2 Capítulos de libro

Título 1: Desarrollo de un prototipo educativo web y móvil para la promoción de la identidad cultural y gastronómica

Autores: Pablo Alejandro Quezada-Sarmiento, David Gonzalo Rodas-Macas, Cristóbal Suárez-Guerrero, Patricia Marisol Chango Cañaveral, Lizbeth Ximena Suárez -Morales

Aporte en la tesis doctoral: Uso de herramientas de CN en el contexto de la enseñanza y aprendizaje de la programación web y móvil en alumnos de informática. Es resultado es derivado de los casos cuasiexperimentales.

Estado: Publicado

Base de Datos de Indexación: THOMSON REUTERS en SPI Scholarly Publishers Indicators (Q1).

Cita: Abreu, António, Castanho, Rui Alexandre, Mora Aliseda, Julián, Vidal Carvalho, Joao (2022). *Transformación Digital e Innovación Tecnológica en la Educación*. Aranzadi

Link: <https://www.dykinson.com/libros/transformacion-digital-e-innovacion-tecnologica-en-la-educacion/9788413915654/>

Título 2: Use of cloud computing tools on pedagogical and educational contexts

Autores: Pablo Alejandro Quezada-Sarmiento Cristóbal Suárez-Guerrero, Magaly Margarita Narvaez-Ríos y Luis Germánico Gutiérrez -Albán

Aporte en la tesis doctoral: Validación del uso de herramientas de computación en la nube, propuesta metodológica y pedagógica de enseñanza. Propone una metodología pedagógica que permite el uso de herramientas de computación en la nube como material complementario en el desarrollo de habilidades computacionales.

Estado: Indexado

Base de Datos de Indexación: Scopus -WOS – Capítulo de Libro Springer

Cita: Quezada-Sarmiento, P.A., Suárez-Guerrero, C., Narvaez-Ríos, M.M., Gutiérrez-Albán, L.G. (2022). Use of Cloud Computing Tools on Pedagogical and Educational Contexts. In: Rocha, A., Adeli, H., Dzemyda, G., Moreira, F. (eds) Information Systems and Technologies. WorldCIST 2022. Lecture Notes in Networks and Systems, vol 468. Springer, Cham. https://doi.org/10.1007/978-3-031-04826-5_35

Link: https://doi.org/10.1007/978-3-031-04826-5_35

Título 3: Libro de Actas: XXX JUTE Tenerife 2023 (Jornadas Internacionales Universitarias de Tecnología Educativa)

Título: Mobile-D y Computación en la Nube en la enseñanza

Autores: Pablo Alejandro Quezada-Sarmiento, Cristóbal Suárez-Guerrero

Aporte en la tesis doctoral: Validación del uso de herramientas de computación en la nube, propuesta metodológica y pedagógica de enseñanza.

Estado: Publicado

Cita: Libro de Actas de las XXX Jornadas Internacionales de Tecnología Educativa: 30 años de docencia e investigación en Tecnología Educativa: Balance y futuro / coord. por Manuel Area Moreira, 2023, ISBN 978-84-09-55300-6, págs. 315-317

Link: <https://manarea.webs.ull.es/libro-de-actas-de-las-xxx-jornadas-internacionales-universitarias-de-tecnologia-educativa-jute-tenerife-23/>

1.6.3 Artículos publicados e indexados en eventos científicos

Artículo 1. Integración de Herramientas de Computación de la Nube y Cuerpos de Conocimientos para la Gestión de Proyectos de Programación.

Autores: Pablo Alejandro Quezada-Sarmiento, Santiago Mengual-Andrés, Liliana Enciso, Wilmar *Hernández*

Aporte en la tesis doctoral: El artículo publicado permitió evaluar el uso de herramientas de CN y el desarrollo de casos experimentales en el contexto de la enseñanza de la Programación y Gestión de Proyectos de Software, que estuvieron alineados a los estándares y currículo de la ingeniería informática.

Evento: 2018 13th Iberian Conference on Information Systems and Technologies (CISTI)

Estado: Indexado

Base de Datos de Indexación: Scopus

Cita: Quezada-Sarmiento, P. A. Enciso, L. Mengual-Andrés S. and Hernandez, W. "Integration of cloud computing tools and knowledge bodies for the management of programming projects," *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)*, Caceres, Spain, 2018, pp. 1-4, DOI: 10.23919/CISTI.2018.8399219

Link: <https://ieeexplore.ieee.org/document/8399219>

Artículo 2. Cloud computing and affective computer approach on development programming skills

Autores: Pablo Alejandro Quezada- Sarmiento, Santiago Mengual -Andrés

Aporte en la tesis doctoral: Descripción de la propuesta de la tesis doctoral y empleo de las herramientas de computación en la nube para el proceso de enseñanza y aprendizaje de la programación en ingenieros informáticos.

Evento: 14th Iberian Conference on Information Systems and Technologies, CISTI 2019

Estado: Indexado

Base de Datos de Indexación: Scopus

Cita: Quezada- Sarmiento P. A. and Mengual-Andrés S., "Cloud Computing and Affective Computer Approach on development programming skills," *2019 14th Iberian Conference on Information Systems and Technologies (CISTI)*, Coimbra, Portugal, 2019, pp. 1-5, DOI: 10.23919/CISTI.2019.8760697.

Link: <https://ieeexplore.ieee.org/document/8760697>

En la tabla 2 se presenta el resumen de las publicaciones generadas en el desarrollo de la presente tesis doctoral.

Tabla 2*Resumen de Producción Científica generada durante el desarrollo de la tesis doctoral*

Tipo	Tema	Base de Datos	Cuartil-Categoría	Año de Publicación
Capítulo de Libro	Mobile-D y Computación en la Nube en la enseñanza	Libro de actas JUTE	N/A	2023
	Use of cloud computing tools on pedagogical and educational contexts.	Scopus	Q4	2022
	Desarrollo de un prototipo educativo web y móvil para la promoción de la identidad cultural y gastronómica.	Thomson Reuters	SPI Scholarly Publishers Indicators (Q1).	2021
	Cloud computing in the training process in web programming.	Scopus	Q4	2021
	Principles of the semantic web and affective computing in a sustainable ecotourism through the development of educational web application.	Scopus	Q3	2019
Artículos de Revistas	Tourism and Gastronomy Management through an Educational Web Platform on Yacuambi of the Province of Zamora Chinchipe -Ecuador.	Scopus	No aplica	2020
Artículos de conferencias	Cloud computing and affective computer approach on development programming skills.	Scopus	Core Rank C	2019
	Integración de Herramientas de Computación de la Nube y Cuerpos de Conocimientos para la Gestión de Proyectos de Programación.	Scopus	Core Rank C	2018

Nota. Detalles de la producción científica generada en el proceso de desarrollo de la tesis doctoral.

Capítulo 2: La dimensión tecnológica

La Organización para la Cooperación y el Desarrollo Económico señala que el contexto educativo global ha cambiado rápidamente, impulsado por dos fenómenos principales: el fortalecimiento de la economía del conocimiento, que trae importantes beneficios para las personas que apuestan a la educación de calidad, a los países que los incentivan y, el segundo, el crecimiento de la cobertura en educación superior, que aumentó la oportunidad de millones de personas (OCDE, 2012).

El fortalecimiento de la economía del conocimiento implica que las habilidades, el conocimiento y la educación son cruciales para el crecimiento económico y el desarrollo personal. En este contexto, la educación de calidad se convierte en una inversión estratégica que puede generar beneficios significativos, tanto para las personas como para las naciones en términos de mayor productividad, innovación y competitividad. Por otro lado, el aumento de la cobertura en educación superior significa que más personas tienen acceso a la formación académica y profesional necesaria para participar efectivamente en la economía globalizada. Esto no solo es fundamental para cerrar brechas de desigualdad, sino también para fomentar una fuerza laboral más capacitada y diversa.

Estos cambios señalan la importancia de políticas educativas que promuevan la calidad y la accesibilidad a la educación, para asegurar que los beneficios de la economía del conocimiento sean ampliamente compartidos y que el potencial humano de las sociedades sea plenamente aprovechado.

Para Blanco y Blanco (2007) argumenta:

La educación universitaria es relevante para el funcionamiento de la sociedad actual y, como tal, su calidad es un reto y una función que debe ser asumida por las universidades. Para cumplir su función debe tener presente la calidad en la docencia, la investigación y la extensión académica. Para alcanzar y mantener un alto nivel de calidad, las IES deben enfocarse en tres áreas: la docencia, la investigación y la extensión académica (p.125).

1. Calidad en la docencia: Esto implica asegurar que los métodos de enseñanza sean efectivos y que los profesores estén bien capacitados y actualizados en sus campos de especialización. La formación continua del profesorado y la evaluación sistemática de los métodos pedagógicos son esenciales para mantener un alto estándar en la enseñanza.
2. Calidad en la investigación: Las IES deben fomentar un ambiente que valore y promueva la investigación. Esto incluye proporcionar recursos adecuados: financiación, infraestructura y acceso a tecnologías avanzadas, además de un clima que estimule la colaboración interdisciplinaria y la innovación.
3. Calidad en la extensión académica: La extensión académica se refiere a las actividades que prolonguen el impacto de la universidad más allá del campus, incluyendo la colaboración con la comunidad, la industria y otras instituciones educativas. Esto asegura que el conocimiento generado en la universidad tenga un impacto directo y práctico en la sociedad.

Por lo tanto, según Blanco y Blanco (2007), “para que las IES cumplan con su rol dentro de la sociedad, deben adoptar un enfoque integral, no solo centrado en la enseñanza, sino que promueva la investigación y la extensión académica como pilares de la excelencia educativa”(p.126).

La calidad educativa es un concepto con muchas aristas y definiciones, de acuerdo con Acevedo (2017) la calidad educativa, como término difuso, tiene diferentes perspectivas que no permiten dar una definición única. Sin embargo, se puede mencionar con certeza que la calidad se consolida como una cultura en las IES, que se debe vincular con el avance en las TIC, de tal forma que el aprendizaje y la enseñanza se basen en Internet. El aprendizaje electrónico (e-learning) es una de las respuestas más poderosas a estos cambios tecnológicos crecientes en los sectores educativos (Baelo,2009). Su impacto educativo y sus beneficios deben de ser enfocados para lograr que el aprendizaje sea accesible, asequible,

disponible y conveniente. En esta línea hay dos conceptos que llaman especialmente el interés: la CN y CA, materia de este capítulo que delimita la dimensión tecnológica de la tesis doctoral.

2.1 Computación en la Nube (CN)

En esta sección se presentan las diferentes conceptualizaciones, referentes a la CN en los contextos tecnológicos y educativo, que serán la base para comprender su importancia y aplicabilidad en el modelo pedagógico planteado en la presente tesis doctoral.

2.1.1 Conceptualizaciones de CN

De acuerdo con Dekson et al. (2009):

El uso de la tecnología en la nube ha hecho que el mundo de la educación sea más integrado, en red y compuesto. Los avances tecnológicos actuales desempeñan un papel clave en el desarrollo del sector de la educación formal, al ofrecer múltiples formas de aprendizaje y comunicación que pueden satisfacer la demanda de educación continua a un bajo costo. Algunas de las ventajas y cambios más destacados del uso de la CN en el contexto educativo son los siguientes (p.155).

1. **Accesibilidad mejorada:** La CN permite que los recursos educativos estén disponibles en cualquier momento y lugar. Esto es beneficioso especialmente para estudiantes que no tienen fácil acceso a bibliotecas físicas o recursos educativos. Con solo una conexión a internet, tienen disponible una vasta cantidad de recursos desde cualquier dispositivo.
2. **Costos Reducidos:** Los sistemas basados en la CN reducen la necesidad de hardware costoso y mantenimiento de infraestructura de TI para las instituciones educativas. Esto se traduce en menores costos de matrícula y de otros gastos para los estudiantes, haciendo la educación accesible a un público más amplio.
3. **Aprendizaje personalizado:** Las plataformas de aprendizaje, basado en CN, pueden adaptar los materiales y recursos a las necesidades individuales de los estudiantes, permitiendo un aprendizaje personalizado. Esto se logra a través del uso de algoritmos que ajustan el contenido según el progreso y estilo de formación del estudiante.

4. Colaboración facilitada: La CN fomenta mayor colaboración entre estudiantes y profesores, incluso en entornos virtuales. Herramientas como Google Drive o Microsoft Teams permiten que los usuarios compartan documentos y colaboren en tiempo real, lo que mejora la comunicación y el flujo de trabajo en proyectos grupales.
5. Continuidad y expansión educativa: La CN facilita la educación continua y a distancia, permite a los estudiantes seguir aprendiendo sin importar sus circunstancias personales o globales, como se ha visto durante la pandemia de COVID-19. También hace posible la expansión de programas educativos que pueden ofrecerse a escala global sin la necesidad de infraestructura física adicional.

A continuación, se presenta la información relevante que contextualiza la investigación planteada en la presente tesis doctoral y facilita su comprensión. Las diferentes conceptualizaciones referentes a la CN, permitirán entender sus características y aplicabilidad en diferentes ámbitos, particularmente en el educativo.

Según Sultan (2010) el término “nube” probablemente se inspiró en las ilustraciones de los libros de texto de Tecnologías de la Información (TI) que representaban entornos remotos como internet a manera de imágenes en la nube; que buscaban ocultar la complejidad que se esconde detrás de ellos.

De acuerdo con *Microsoft* (2020):

La CN es el suministro de servicios informáticos (servidores, almacenamiento, bases de datos, redes, software, análisis e inteligencia) a través de “la nube” (Internet), cuyo objetivo es ofrecer una rápida innovación, recursos flexibles y economías de escala. Lo habitual es pagar solamente por los servicios utilizados en la nube, de tal forma que esto ayude a reducir los costos operativos para ejecutar la infraestructura con más eficacia y escalar a medida que cambian las necesidades del negocio. Así mismo, la CN es un nuevo paradigma informático, emergente para la prestación de servicios informáticos que se basa en una serie de tecnologías existentes, por ejemplo, Internet, virtualización, computación grid, servicios web, entre otras.

La CN, como se indicó anteriormente, es un paradigma emergente de servicios de computación y, como otros servicios nuevos de esta escala y complejidad, es probable que haya temores, incertidumbres y preocupaciones sobre la madurez de la tecnología. Las preocupaciones más importantes pueden relacionarse con el control, el bloqueo del proveedor, el rendimiento, la latencia, la seguridad, la privacidad y la fiabilidad (Sultan, 2010).

La CN se ha convertido en un modelo de computaciones populares para apoyar el procesamiento de grandes volúmenes de datos, utilizando clústeres de computadores (Olugbenga et al., 2018).

La CN es una arquitectura basada en Internet que crea un entorno informático para proporcionar disponibilidad, escalabilidad y flexibilidad de las infraestructuras informáticas a diferentes niveles de abstracción. Es una de las nuevas tendencias que tienen un significativo impacto en el entorno de enseñanza y aprendizaje; por ello, se plantea el uso de estas herramientas en el ámbito educativo, aplicado en el contexto de la CA (Hayes, 2018,p.10).

Más ampliamente, Taurion (2009) define a la CN como cualquier servicio que no se ejecuta en la computadora del usuario, pero al que se puede acceder desde cualquier lugar sin la necesidad de instalar aplicaciones en su máquina ni pagar licencias de software.

La CN permite que los “activos digitales” se transfieran gradualmente a los principales “centros de datos virtuales”, a través de servicios prestados por empresas como IBM, Dell, Sun, Google, Accenture, Microsoft y otras, que asumen la responsabilidad del almacenamiento y operación. El cloud computing, en español CN, es una tecnología cada vez más utilizada en diferentes campos y disciplinas, por ser una tendencia relativamente nueva, sobre todo en el desarrollo de la industria TIC, centrada principalmente en los usuarios e impulsada por el creciente uso de dispositivos móviles (Mejía y Ballesteros, 2014,p.130).

Para los profesionales de TI, la CN es un nuevo modelo de negocio y una plataforma tecnológica para el desarrollo y despliegue de aplicaciones, para los usuarios finales es una nueva y económica forma de emplear sus recursos (Bronzin, 2009). En la educación superior,

los servicios de CN se usan comúnmente para proporcionar a los estudiantes los medios, en los cuales colaboren e interactúen en un espacio de aprendizaje distribuido. La CN es un nuevo servicio accesible en Internet. Se puede definir como un nuevo modelo informático, en el que los recursos físicos y locales se convierten en recursos escalables, también disponibles de forma virtual.

Vorobeva et al., (2022) manifiesta que:

Las principales ventajas de migrar a CN están asociadas a la facilidad de desarrollo, reducción de los costos operativos y mejora en el rendimiento, con énfasis en la interoperabilidad: 1. Capacidad de desarrollo en diferentes plataformas de software; 2. Alta capacidad informática, capacidad HPC (informática de alto procesamiento); 3. Capacidad de seguridad y almacenamiento, con datos en servidores en la nube; 4. Reducción de costos operativos, pagando por uso y no por licencias; 5. Transformación de costos fijos en costos variables, uso e instalación rápida del sistema previo pago (p.164)

En términos de evolución de las tecnologías de la información, la CN es tanto o incluso más revolucionaria que los grandes saltos tecnológicos del pasado, como el tiempo compartido, que en la década de 1970 permitió que los grandes ordenadores ejecuten varias tareas al mismo tiempo, aumentando su procesamiento. El Instituto Nacional de Estándares y Tecnología definió la CN como un modelo para facilitar el acceso a las redes bajo demanda y, de este modo, combinar los valores de configuración de los recursos computacionales (como redes, servidores, medios de almacenamiento, aplicaciones y servicios) que se pueden configurar y liberar rápidamente con una gestión o interacción mínima con el proveedor de servicios (Mell y Grance, 2009).

La CN ofrece oportunidades, pero también algunas limitaciones, por el hecho de que todos los datos y aplicaciones de este tipo de tecnología son almacenados en algún servidor o sitio en internet.

Para los usuarios habituales de Internet y computación, la CN es cualquier actividad que se realice en línea, como el acceso a los datos o el uso de un programa de software

desde diferentes dispositivos, independientemente de la manera en que se efectúa el acceso a Internet (Horrigan, 2008). Se puede decir que la computación en la nube es el nuevo conductor de la revolución en TI, en ella se están desarrollando nuevos servicios, cambiando la forma de acceso a la información y su uso, así como, el mantenimiento en conjunto con la financiación de los servicios (Hewitt, 2008).

La CN tiene el potencial para crear nuevas interacciones, metáforas y maneras de pensar sobre el aprendizaje, su construcción y la adquisición de conocimientos a partir de las experiencias (Kop, 2011).

Zou et al. (2022) manifiestan que la CN es un modelo de red de acceso bajo demanda, que permite compartir grupos de recursos informáticos configurables. En comparación con las arquitecturas de servicios convencionales, la computación en la nube presenta nuevos desafíos de seguridad en la gestión y control de servicios, protección de privacidad e integridad de los datos en sus respectivas bases distribuidas, la copia de seguridad de datos y la sincronización (p.8).

Wei (2022) manifiesta que la CN es una nueva plataforma para la educación en programación, dirigida a estudiantes universitarios contemporáneos. Esta plataforma se basa en la construcción de teoría, aprendizaje cooperativo y comunidad de aprendizaje, y es de gran ayuda para mejorar la calidad de la enseñanza de la Programación de Estudiantes Universitarios (PEU) (p.257).

Vorobeva et al. (2022) expresan que:

La idea de la CN no es nueva, pues ya se ha desarrollado y discutido durante muchos años. La CN es un modelo que permite obtener acceso a la red, por pedido del conjunto de servicios computacionales ajustables (como infraestructura, aplicaciones y almacenamientos). Los servicios en la nube, y los productos de almacenamiento de datos, permiten a sus usuarios almacenar y compartir todo tipo de documento y archivo desde cualquier dispositivo conectado a Internet. Existen varios servicios en la nube que se pueden subdividir: SaaS (Software as a Service), PaaS (Platform as a Service), IaaS (Infrastructure

as a Service). Además, hay modelos de despliegue, como nube pública, residencial, híbrida o comunitaria (p.165).

Con las definiciones previamente descritas, nuestro enfoque de CN se orientó al uso de estas tecnologías en el contexto educativo con la finalidad de proponer un modelo pedagógico que aporte al desarrollo de las destrezas de programación en el área de las ciencias de la computación y afines.

A continuación, se detallan los tipos de CN y sus principales características.

2.2 Tipos de Computación en la Nube

De acuerdo con Slama et al. (2010) la popularización de la CN inspiró la aparición de muchos nuevos proveedores de este tipo de servicios. La gran cantidad y disponibilidad de proveedores de nube lleva a los usuarios a una elección compleja o incluso poco práctica del más adecuado para satisfacer sus necesidades sin automatización. El problema de la selección del proveedor de la nube aborda ese dilema. Habría que decir también que Zou et al.(2022) expresan que la CN es un modelo de red de acceso bajo demanda para compartir grupos de recursos informáticos configurables especialmente en el campo educativo.

Polze (2009) aduce que las instalaciones de CN se pueden clasificar de acuerdo con tres estrategias generales:

Infraestructura como servicio (IaaS) o subcontratación de equipos utilizados en operaciones, incluidos almacenamiento, servidores y componentes de red, como Amazon Web Services y VMware iCloud.

Plataforma como servicio (PaaS) o entrega de sistemas operativos y servicios asociados, a través de Internet, sin descarga o instalación. Se debe agregar que los recursos de la computadora personal se transportan a Internet, desde donde se ponen a disposición de los clientes y usuarios, por ejemplo, Google App Engine y Windows Azure Platform,

Software as a Service (SaaS) donde todo el software, incluidos el sistema operativo y los datos se implementan en la nube: aplicaciones de productividad Microsoft Office Live y Microsoft Business Productivity Online Standard Suite; gestión de figuras (Flickr); correo electrónico (Google Mail) y almacenamiento (Dropbox, Microsoft Mesh, SkyDrive, Gladinet).

Esta es una clasificación didáctica, a diferencia de la práctica, en la que los proyectos en la nube pueden usar una combinación de estas tres estrategias con el objetivo de obtener el mejor beneficio técnico y económico en términos de hardware, y flexibilidad para satisfacer la demanda, a través de una estructura de pago asociada a la utilización. Así mismo, los entornos de CN también se pueden clasificar de acuerdo con el acceso y la ubicación que definen las nubes públicas, disponibles para cualquier usuario de internet, ya que Amazon Elastic Computer Cloud, Google App Engine y Microsoft Windows Azure pueden acceder y emplear una nube pública; así como también nubes privadas: entornos de nube personales (con número limitado de usuarios) y nubes híbridas, también conocidas como "nubes privadas virtuales" con servicios ejecutados en una nube pública, pero dentro del límite de VPN (red privada virtual).

2.2.1 Tipos de CN según la propiedad del servicio

De acuerdo con Xue et al. (2022) el rápido desarrollo de internet y los dispositivos móviles dan como resultado un modelo comercial de *crowdsensing*, en el que las personas (usuarios) están dispuestas a contribuir con sus datos para ayudar a la institución (recolector de datos) a analizar y publicar información útil. Aquí, la CN juega un papel trascendental para la gestión de los datos que se generan (p.10). A continuación, se detallan los tipos de servicios de computación en la nube, basados en las propiedades del servicio.

Nube pública. – Generalmente, este servicio de CN es ofrecido por empresas externas, como *Google* o *Amazon*, que se encargan de su gestión y mantenimiento. Esta nube tiene algunas implicaciones, entre las que destacan:

1. La información se almacena en servidores de terceros, de los que no somos propietarios.

2. Los recursos de los servidores se comparten entre todos los clientes del proveedor, es decir, nuestros datos y los de otras empresas podrían estar alojados en el mismo equipo.

3. No requiere de una inversión inicial en infraestructura, ya que pertenece al proveedor.

4. El mantenimiento y actualización corre a cargo de la empresa que contratamos (beServices, 2018).

Hay que mencionar que las soluciones de implementación de clústeres informáticos a gran escala existentes suelen utilizar la instauración de un código fuente u otras herramientas. El gran desafío de los métodos de implementación existentes es reducir el tiempo excesivo de distribución de imágenes y evitar defectos de configuración (Jiang et al., 2022,p.1822). En contraste, Hofmann y Woods (2010) plantean que el modelo de CN, especialmente la nube pública, no es adecuado para muchas aplicaciones comerciales y es probable que no lo siga siendo durante muchos años, debido a limitaciones fundamentales en la arquitectura y el diseño.

Nube privada. - Al contrario que el caso anterior, se implementa la solución de CN en infraestructura propia. Las características principales de este tipo de CN son:

1. Los servidores e infraestructura física son de propiedad del usuario.
2. Ofrece un mayor grado de seguridad, ya que los datos no son gestionados por terceros.
3. Hardware dedicado y adaptado a las necesidades del usuario.
4. La inversión inicial es superior.
5. Los costes de mantenimiento y actualización son superiores a una solución pública (beServices, 2018).

Nube híbrida.- Es una combinación de los casos anteriores, en los que se dispone de servidores propios y se utilizan, en casos puntuales, la nube pública para funcionalidades

específicas o conexiones con herramientas externas. Shi et al. (2022) manifiestan que la nube múltiple se está convirtiendo en un ecosistema de nube popular porque permite a los usuarios empresariales compartir la carga de trabajo entre múltiples proveedores de servicios en la nube para lograr servicios de alta calidad con costos operativos más bajos y una mayor resiliencia de las aplicaciones. En multinube, los servicios están ampliamente distribuidos en diferentes ubicaciones con precios diferenciados (p.1986).

Nube comunitaria (community cloud).- Los servicios se comparten en una comunidad cerrada de entidades que tienen los mismos objetivos, de manera que colaboran entre ellas. En este caso, la nube es gestionada y administrada por una o más entidades de la comunidad (beServices, 2018).

De acuerdo con Fahmideh et al. (2022) las plataformas en la nube implican diferentes actividades interrelacionadas: planificación, diseño de la arquitectura, realojamiento, refactorización de código y otras actividades. En este sentido, la literatura de CN ha visto el surgimiento de diferentes métodos con un punto de vista dispar del mismo proceso de reingeniería de aplicaciones heredadas, subyacente a las plataformas en la nube, en especial en las nubes comunitarias (p.5).

2.2.2 Según el modelo de servicio

Los modelos de CN se basan en el paradigma de procesos modernos, que ofrecen nuevas alternativas a las empresas de diversos rangos para la implementación de modelos de negocios innovadores. Con la ayuda de estos nuevos modelos de negocios, las pequeñas empresas podrán utilizar plataformas de CN y aumentar gradualmente su capacidad de cómputo y almacenamiento de datos según los requisitos en modo de tiempo real, lo que crea una oportunidad única para la competencia en el mercado (Vorobeva et al., 2022). A continuación, se detallan las características de los servicios de CN de acuerdo con el modelo de servicio.

Software as a Service (SaaS).

Chen et al. (2022) argumentan que la CN se está volviendo cada vez más popular, por ser un servicio nuevo y moderno que puede cumplir con los requisitos esperados por los clientes de manera efectiva. Los proveedores de servicios en la nube juegan un papel importante en la construcción de las plataformas informáticas, que permiten a los clientes disfrutar de los servicios sin prestar atención a su proceso de ejecución (p.205). Así mismo, en Sultan (2010) se explica el software como servicio (SaaS): en esta capa, las aplicaciones se entregan a través de Internet como un servicio, en lugar de instalar y mantener software, liberándose de la gestión compleja de software y hardware.

Plataformas a Service (PaaS)

Zhang et al. (2022) manifiestan que el modo de procesamiento de información tradicional basado en una estación maestra de nube única tiene algunas desventajas: alto retraso en la transmisión de datos, alta congestión, información lejos del equipo y el procesamiento lento de las colas.

Dentro de este mismo orden de ideas, para comprender la Plataforma como servicio (PaaS), es necesario recordar el modelo de computación tradicional, en el que cada aplicación administra el hardware requerido localmente, el sistema operativo, la base de datos, el middleware, los servidores web y otro software.

Infrastructure as a Service (IaaS).

Al-Wesabi et al. (2022) manifiestan:

Que la CN se ha convertido rápidamente en un marco efectivo para ofrecer infraestructura, recursos y servicios de TI en una base de pago por uso. Los productos ofrecidos a través de este método incluyen la entrega remota (a través de internet) de una infraestructura informática completa (computadoras virtuales, servidores, dispositivos de almacenamiento, etc.). Las soluciones de IaaS proporcionan a las empresas opciones de almacenamiento, servidores y redes que no requieren grandes salas de servidores que ocupen espacio y energía. Uno de

los motivos clave por los que las empresas optan por soluciones de IaaS es reducir la inversión de capital y transformarla en gastos operativos (p.4).

Así mismo, Chakravarthi et al. (2022) manifiestan que el aprovisionamiento de recursos y la ejecución del flujo de trabajo en un entorno de varias nubes, con un marco de pago por uso, han captado recientemente la atención de la comunidad de investigación de la CN. Debido a la dinámica de la nube, la programación de flujos de trabajo en la plataforma multinube es un desafío. En particular, con los tipos de recursos heterogéneos, los múltiples mecanismos de facturación, la elasticidad, el aprovisionamiento bajo demanda y la confiabilidad de los sistemas (p.1202).

2.3 Soluciones tecnológicas actuales de CN

Núñez et al. (2022) argumentan que, durante la última década, la comunidad investigadora ha desarrollado diferentes herramientas de simulación para modelar y estudiar sistemas de nubes. Sin embargo, los simuladores de nube actuales se enfocan en características específicas que normalmente no cubren completamente todos los aspectos de la infraestructura de la nube. En este sentido, Rahim et al. (2019) manifiestan que la CN permite utilizar activos informáticos y de capacidad en una premisa medida, también hace posible reducir los intereses en una base de procesamiento de asociaciones. Es por eso que en el ámbito educativo se están adoptando activamente tanto tecnologías digitales como tecnologías en la nube, análisis de datos, plataformas digitales, dispositivos móviles, entre otras.

De acuerdo con Aslakhanova et al. (2021), las tecnologías digitales permiten que una IES venda productos educativos con costos mínimos en el menor tiempo posible, expanda el mercado de productos, analice las necesidades de la población y amplíe la base de clientes (futuros estudiantes) de acuerdo con el contexto, en el que se desarrolle (p.6).

A continuación, se presentan algunas soluciones tecnológicas respecto a la CN.

2.3.1 Amazon Web Services

Según Russo y Ellabidy (2013), Amazon ofrece infraestructura en la nube a través de sus Amazon Web Services (AWS). Se debe agregar que cientos de miles de clientes se han unido a la comunidad de Amazon Web Services para crear soluciones para sus negocios. La plataforma de CN de AWS brinda flexibilidad para crear la aplicación a su manera, independientemente de su caso de uso o industria (Robinson, 2008). Se debe agregar que Amazon ofrece varias opciones diferentes para implementar cargas de trabajo en AWS en forma de instancias bajo demanda, reservadas y puntuales (Mandal et al., 2022).

Los servicios *web* de *Amazon* (AWS) también ofrecen muchas herramientas disponibles para los desarrolladores (Gilani et al., 2015). Habría que decir también que *Amazon EC2* tiene algunas ventajas: la informática elástica a escala *web*, los servicios de alojamiento en la nube completamente controlados y flexibles, además de un entorno integrado, fiable y seguro.

2.3.2 Google Cloud Platform

Google Apps es uno de los mayores proveedores de servicios de CN, pues ofrece múltiples instalaciones de colaboración. Las adaptaciones medibles y planificadas a Google Apps pueden proporcionar muchos beneficios para las instituciones de educación superior. En este mismo orden de ideas, Google Apps for Education es un servicio que proporciona varias versiones que se pueden personalizar con determinados productos de Google, aunque utiliza su propio nombre de dominio (Alim y Jin, 2017,p.74).

2.3.3 Microsoft Azure

Hoy en día, la computación de alto rendimiento (HPC) se entrega como un servicio en la nube. Microsoft Azure, uno de los principales proveedores de servicios en la nube, tiene una clase designada de máquinas virtuales (VM) que se recomienda para los programas de HPC (Aljamal et al., 2020, pp. 78).

De acuerdo con Microsoft (2022), Windows Azure es la plataforma de Microsoft que funciona como una oferta de servicio y que además incluye un sistema operativo y un conjunto de servicios para desarrolladores que proporcionan la funcionalidad para crear aplicaciones que abarcan los escenarios de Web a una empresa de consumidor. Windows Azure y SQL Azure son los componentes clave de esta plataforma. En el mismo orden de ideas, Windows Azure proporciona capacidades de proceso a los desarrolladores: almacenamiento, redes, entrega de contenido y middleware a petición para hospedar, escalar y administrar aplicaciones en Internet a través de centros de datos de Microsoft.

Para crear aplicaciones y servicios en esta plataforma, los desarrolladores pueden utilizar su experiencia existente con herramientas de programación estándar. Azure admite estándares, protocolos, lenguajes y entornos de desarrollo populares, incluidos SOAP, REST, XML, NET Framework, C, Java, PHP y Ruby (Khalidi, 2011).

2.3.4 Docker

Docker, como solución de contenedor principal, adopta el mecanismo de copia en escritura (CoW) en sus controladores de almacenamiento. Este mecanismo satisface la necesidad de que diferentes contenedores compartan la misma imagen. Sin embargo, cuando un solo contenedor realiza operaciones como la modificación de un archivo de imagen, se crea un duplicado en la capa superior de lectura y escritura, lo que contribuye a la sobrecarga del tiempo de ejecución. Docker es un software de TI, es una tecnología de creación de contenedores que permite la creación y el uso de contenedores de Linux Red Hat Inc. (2020); en el mismo contexto, Merkel (2014) manifiesta que Docker es una plataforma de contenedor de código abierto para desarrollar, implementar y ejecutar aplicaciones lanzadas por Docker. Inc. en marzo de 2013.

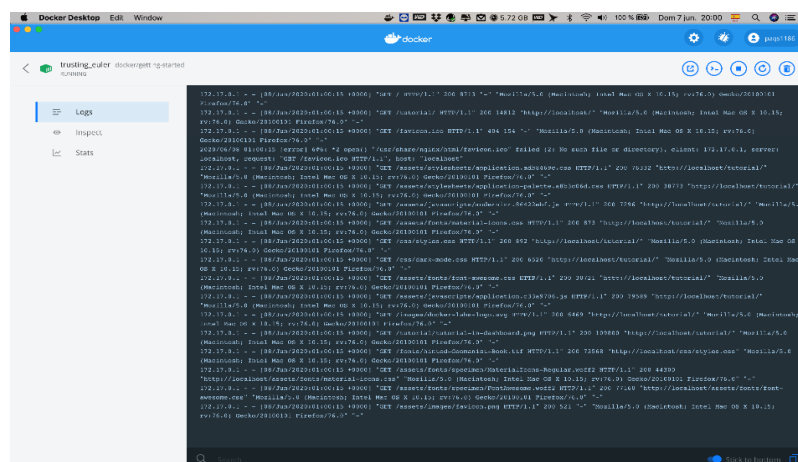
Con Docker, se pueden usar los contenedores como máquinas virtuales extremadamente livianas, modulares y flexibles, puede crearlos, implementarlos, copiarlos y moverlos de un entorno a otro, lo cual le permite optimizar sus aplicaciones para la nube (Red Hat Inc., 2020).

Desde la implementación de la plataforma Docker, han surgido varios proyectos de investigación y utilización de contenedores en tecnologías de virtualización. Hoy en día, los contenedores Docker se utilizan ampliamente para aprovisionar varias aplicaciones a través de hosts físicos compartidos en una forma más ligera que una plataforma de máquina virtual (VM) tradicional (Shu et al., 2017).

Docker ofrece a los usuarios un acceso sin precedentes a las aplicaciones, así como la capacidad de implementar y controlar rápidamente las diferentes versiones y su distribución. Entre las ventajas de los contenedores Docker están: 1. Modularidad, 2. Control de versiones de imágenes y capas, 3. Restauración y 4. Implementación rápida (Jiang et al., 2021). Es importante considerar que Docker crea una forma cuantitativa de una imagen de Docker, que se puede compartir y distribuir fácilmente a través de un repositorio de imágenes (Docker Hub, Gitlab, Quay.io, entre otras). El rápido intercambio y despliegue de imágenes Docker tiene la ventaja de permitir a los desarrolladores compartir una variedad de entornos de implementación de aplicaciones en tiempo real (Kwon y Le, 2020 p.42667).

En la Figura 6 se presenta la interfaz de Docker en escritorio, utilizada en la presente investigación.

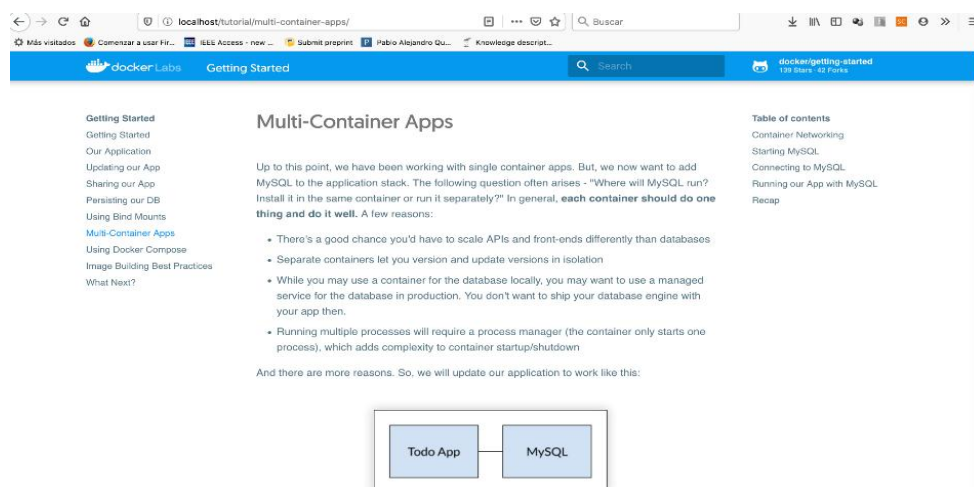
Figura 6
Docker versión escritorio



Nota. Docker Escritorio utilizado en la tesis doctoral

En la Figura 7 se puede visualizar el Docker en Línea implementado en el desarrollo de la presente tesis doctoral.

Figura 7
Docker versión Online



Nota. Docker Online utilizado en la tesis doctoral

2.3.5 YellowCircle

El círculo amarillo es un laboratorio gratuito de CN y proporciona sistemas operativos y aplicaciones, bajo demanda, para aprender o enseñar tecnología. Los estudiantes y profesores pueden registrarse aquí y utilizar los recursos para pruebas y desarrollos. Estos recursos incluyen capacidades únicas de CN e infraestructura en la nube. De acuerdo con Group, Y.C (2020), el objetivo de este proveedor es ayudar a todos a aprender tecnología en todas partes con información real y disponible.

2.4 Computación de la nube en la educación

La introducción de las TIC en las aulas pone en evidencia la necesidad de una nueva definición de roles, especialmente, para los alumnos y docentes. Los primeros, gracias a estas nuevas herramientas, pueden adquirir mayor autonomía y responsabilidad en el proceso de aprendizaje, lo que obliga al docente a salir de su rol clásico como única fuente de conocimiento. Esto genera incertidumbres, tensiones y temores; realidad que obliga a una readecuación creativa de la institución escolar.

Según Khalidi (2011) la CN representa un cambio drástico, no solo en términos de la nueva economía que aporta, sino también en términos de las nuevas posibilidades que ofrece. Darse cuenta de todos los beneficios de este paradigma requiere repensar la forma

en que creamos las aplicaciones, en especial en el contexto educativo (p.32). En la misma línea, de acuerdo con Sultan (2010), varios establecimientos educativos reconocen el potencial de la computación en la nube para mejorar la eficiencia, el costo y la conveniencia para el ámbito educativo.

La CN es adecuada para las necesidades de las IES. Su aplicación en estas instituciones proporciona una solución flexible para satisfacer la creciente demanda de cursos en línea y el desarrollo de tecnologías de aprendizaje. En el mismo orden de ideas, el desarrollo del aprendizaje colaborativo se ha convertido en una necesidad para las instituciones de educación superior (Bora y Ahmed, 2013, p.11).

La CN en forma de software como servicio (SaaS) es adecuada para la optimización del aprendizaje colaborativo (Kats, 2010). Así mismo, Sclater (2010) manifiesta que la mayoría de las instituciones educativas ya están utilizando la CN, principalmente para los servicios de correo electrónico. La CN como herramienta educativa utiliza la retroalimentación con los estudiantes, así como las herramientas de almacenamiento de datos en entornos de aprendizaje virtual y se basa en principios de seguridad (Paterson et al., 2021; Mohanty et al., 2021), validación de sus marcos de trabajo (Lambert et al., 2021) y una adecuada implementación de su infraestructura (Lu et al., 2021).

La CN a menudo se asocia con el e-learning y m-learning (Hurwitz et al., 2010). Esto refiriéndose al entorno de e-learning distribuido principalmente en Internet, en el que los recursos educativos abiertos son producidos, investigados y compartidos por participantes de todo el mundo (Kop, 2011). En este contexto, diversas IES han comenzado su movimiento hacia la CN por la trascendencia que tiene en los estudiantes, también se están comenzando a utilizar los servicios de la nube en un nivel básico, con fines de almacenamiento de datos. El uso de servicios y aplicaciones en la nube entre estudiantes y docentes rompe las barreras espacio temporales o de movilidad, ya que sus recursos educativos y aplicaciones están disponibles a través de los dispositivos portables interconectados por medio de la *web*. Por

ejemplo, las clases pueden ser implementadas fuera del campus institucional o facultad y los estudiantes pueden realizar, y entregar, trabajos o tareas desde diferentes sitios.

La CN por su naturaleza y características tecnológicas brinda una base sólida para ser utilizada en el contexto educativo, en especial en el proceso de enseñanza y aprendizaje, por ello en la presente tesis doctoral se toma como uno de los pilares del modelo pedagógico basado en tecnologías a través de CN y CA para el desarrollo de habilidades de programación en ciencias de la computación.

2.5 Herramientas relevantes de Computación en la Nube utilizadas en el contexto de la enseñanza de la Programación en Educación Superior

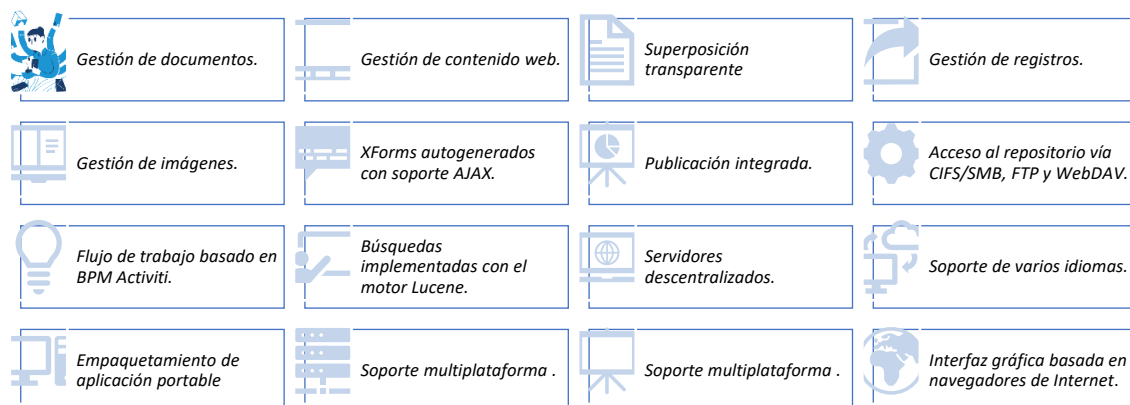
2.5.1 Alfresco

Yan et al. (2014) manifiestan que la herramienta de desarrollo de Alfresco es una plataforma abierta para la gestión de contenido empresarial (ECM) que permite la colaboración y gestión de documentos críticos para la empresa desarrollada en Java, se basa en estándares abiertos y de escala empresarial para sistemas operativos tipo Windows, Unix Solaris y algunas versiones de Linux Formación (p.1605).

De acuerdo con Stratabi (2020), Alfresco está diseñado para usuarios que requieren un alto grado de modularidad y rendimiento escalable. Se debe considerar que Alfresco incluye un repositorio de contenidos, un framework de portal web para administrar y usar contenido estándar en portales, una interfaz CIFS que provee compatibilidad de sistemas de archivos en Windows y sistemas operativos tipo Unix, un sistema de administración de contenido web, capacidad de virtualizar aplicaciones web y sitios estáticos vía Apache Tomcat, búsquedas vía el motor Apache Solr-Lucene y flujo de trabajo en jBPM.

En la Figura 8 se presentan las características de la herramienta Alfresco.

Figura 8
Características de la herramienta de la nube Alfresco

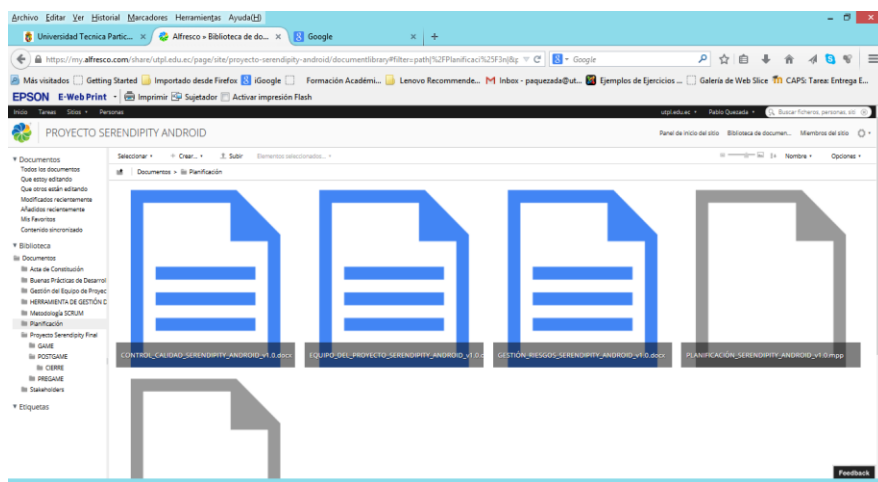


Nota. Características de Alfresco adaptado de Stratabi (2020).

Dentro de las áreas de uso de Alfresco destacan: medicina (Nurhayati y Fitrisari, 2017), gestión administrativa documental (Sousa et al., 2016), gestión empresarial (Yan et al., 2014; Dhouib y Halima, 2013), gestión de servicios y base de datos mediante microservicios (Yan et al., 2014), educación (Quezada et al., 2015; Costagliola et al., 2016; Quezada-Sarmiento et al., 2018; Abeer Yassin et al., 2018; Finogeev et al., 2020), web semántica y colaboración (Janev et al., 2009).

En la Figura 9 se presenta la interfaz gráfica de uno de los proyectos utilizados en el contexto de la presente tesis doctoral.

Figura 9
Interfaz de Alfresco en Proyecto de Software



Nota. Interfaz de la plataforma Alfresco utilizada en la presente tesis doctoral

El uso de Alfresco en la educación, especialmente en el desarrollo de las habilidades de programación en la ingeniería informática, es relevante debido a sus capacidades como sistema de gestión de contenido empresarial (ECM) y gestión de documentos. Alfresco se puede aplicar en varios aspectos del ámbito educativo para mejorar la organización, accesibilidad y colaboración en relación con los documentos y recursos educativos. A continuación, se describen algunas de sus aplicaciones principales:

1. **Gestión Documental:** Alfresco permite a las instituciones educativas organizar eficientemente una gran cantidad de documentos, desde registros académicos hasta publicaciones de investigación. Su capacidad para gestionar versiones y control de acceso ayuda a mantener la integridad y seguridad de los documentos.
2. **Colaboración:** Facilita la colaboración entre estudiantes y profesores que comparten documentos y recursos en un entorno centralizado. Los usuarios pueden trabajar simultáneamente en documentos y guardar los cambios en tiempo real, lo que promueve una interacción más dinámica y eficiente.
3. **Automatización de Procesos:** Alfresco puede automatizar procesos educativos como la admisión de estudiantes, la evaluación y más, mediante flujos de trabajo que reducen la carga administrativa y mejoran la eficiencia operativa.
4. **Integración con Otros Sistemas:** Puede integrarse con plataformas de aprendizaje en línea (como Moodle o Blackboard) y otros sistemas de información estudiantil, lo que permite una experiencia más fluida para los usuarios y una mejor gestión de datos.
5. **Acceso Remoto y Móvil:** Ofrece soluciones de acceso remoto, lo que es especialmente útil en escenarios de aprendizaje a distancia. Los estudiantes y el personal pueden acceder a materiales y colaborar desde cualquier lugar y dispositivo.

2.5.2 Basecamp

Es un software que nació en 2004 con la promesa de facilitar las labores de una compañía importante con listas de tareas por hacer, es un sistema para compartir archivos y manejar documentos con varios miembros de una organización. Es muy útil y sencillo de usar (Franco, 2010). *Base Camp* se basa en la sencillez, claridad y facilidad de uso, esto hace que sea realmente único, ya que aborda la gestión del proyecto desde un ángulo totalmente diferente (con enfoque en comunicación y colaboración) (portalSaS.com, 2012).

Entre las características de BASECAMP destacan:

1. Programar hitos, actividades y entregables.
2. Gestionar varios proyectos a la vez.
3. Distribuir recursos a un proyecto, solo cuando es necesario.
4. Mantener todos los documentos asociados con un proyecto en un solo lugar para facilitar la comunicación entre el equipo (portalSaS.com, 2012).

En el ámbito educativo y en el de salud, BASECAMP permite una adecuada gestión documental de información (Al-Shawwa y Ingram, 2021). En el mismo sentido, Basecamp también proporciona un registro de archivo de actividad de la clase, que se puede utilizar con fines de acreditación y, como tal, sirve como herramienta de evaluación para la experiencia Capstone. El software ofrece a los patrocinadores de proyectos la oportunidad de participar fácilmente ayudando a los estudiantes a administrar el cronograma y el tiempo de los entregables, y sirve como una plataforma para discutir el desempeño y los hábitos de trabajo a lo largo del tiempo. La evaluación del desempeño de los estudiantes en *Basecamp* se realiza de varias maneras: se traza la frecuencia promedio de las entradas de los estudiantes, frente a las indicaciones del profesor de la clase, y se exploran otras medidas de frecuencia de interacción. Además, los estudiantes son entrevistados al final del semestre para conocer sus opiniones y reflexiones sobre la utilidad y facilidad de uso del software (Pezeshki y Racicot, 2006,p.7).

2.5.3 Edmodo

Es una plataforma educativa que funciona igual que una red social, al estilo de Facebook, Tuenti o Twitter. Tiene todas las ventajas de este tipo de webs, pero sin los peligros de las redes sociales abiertas, facilita crear un grupo cerrado entre el alumnado y el profesor para compartir mensajes, enlaces, documentos, eventos, entre otras publicaciones (profesorPaco, 2011). Así mismo, permite crear un espacio virtual de comunicación con el alumnado y otros profesores, en el que se puede compartir mensajes, archivos, enlaces, un calendario de trabajo y proponer tareas, y actividades, para gestionarlas (Unal y Uzun, 2021,p.625).

Mahmud et al. (2021) mencionan que el uso de la aplicación Emodo hace posible un adecuado desarrollo de destrezas y competencias digitales a nivel tecnológico; además, esta herramienta complementaria permite generar un adecuado entorno de aprendizaje. Así mismo, Altunkaya y Ayranci (2020) declaran que Edmodo es una herramienta de CN que permite desarrollar las habilidades de escritura académica y generar un ambiente de aprendizaje dinámico. En el mismo orden de ideas, Alamsyah et al. (2021) proponen el uso de Edmodo como una herramienta para la enseñanza de conceptos en la ciencia. Pérez et al. (2021) proponen el uso de Edmodo para la enseñanza de matemáticas en los colegios de Perú.

2.5.4 Redbooth

De acuerdo con Social Media (2014), Redbooth es una plataforma de colaboración y comunicación que proporciona un único lugar para tareas compartidas, debates, intercambio de archivos, grupos de chat y videoconferencia de alta definición. Redbooth es flexible y fácil de usar, permite realizar su trabajo a los equipos y departamentos de proyectos en miles de empresas.

2.5.5 DotProject

Es una herramienta orientada a la Gestión de Proyectos, la cual consiste en dirigir a la administración de recursos para desarrollar un producto, en cuya elaboración requiera de

un conjunto de actividades o tareas que se desarrollen en forma paralela o independiente. La aplicación, según EcuRed (2020), consta de un conjunto de entidades ordenadas jerárquicamente, que brindan funcionalidad al producto. El software es libre. Además, tiene soporte gratuito provisto por voluntarios en los foros de la comunidad y vía documentación en línea. También es posible contratar soporte para el equipo de desarrollo a través del llamado Foro de Soporte Prioritario.

Entre las características de la herramienta DotProject destacan:

1. Permite la gestión y planificación de proyectos en entornos colaborativos.
2. Permite la participación en línea de los miembros de un proyecto.
3. Permite la asignación de recursos (equipamientos, mobiliario) a un proyecto o varios, así como la descomposición en tareas.
4. Permite clasificar y/u ordenar los proyectos en función de su estado: en curso, pendientes y cerrados.
5. Permite vista de eventos y tareas en calendario, filtrado por estado, actividad y proyecto.

Permite la visualización de informes y estadísticas sobre los proyectos registrados (EcuRed,2020).

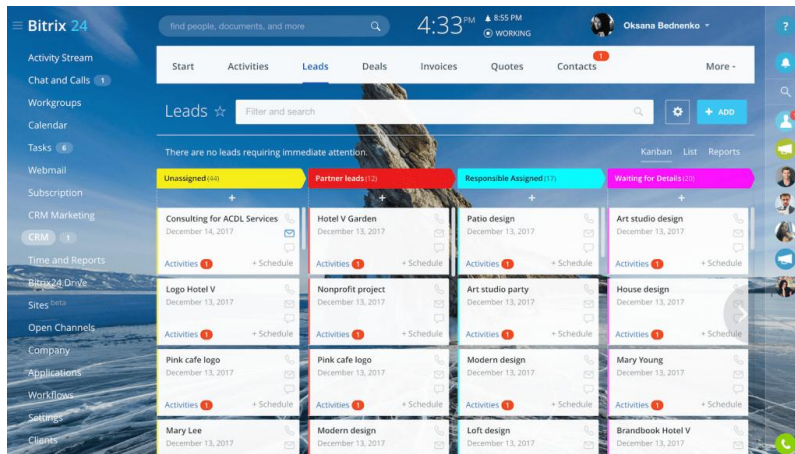
2.5.6 BITRIX24

Es un servicio de software desarrollado por Bitrix, Inc. que proporciona a una empresa una amplia gama de medios de trabajo en equipo y redes sociales. El servicio está destinado principalmente a las pequeñas y medianas empresas (Bitrix Inc, 2020).

Bitrix24 es compatible con absolutamente todos los navegadores web modernos. Algunas características pueden no estar disponibles en los navegadores obsoletos. Por ende, Bitrix24 se basa en Bitrix24 Intranet, lo que significa que será capaz de migrar a un procesador e instalar una versión en cualquier momento, preservando los trabajos de la empresa,

documentos y el historial de las comunicaciones. En la Figura 10 se presenta la interfaz gráfica de Bitrix24.

Figura 10
Interfaz Bitrix



Nota. Interfaz de la plataforma Bitrix adaptado de (Parada Visual , 2019)

2.5.7 Redmine

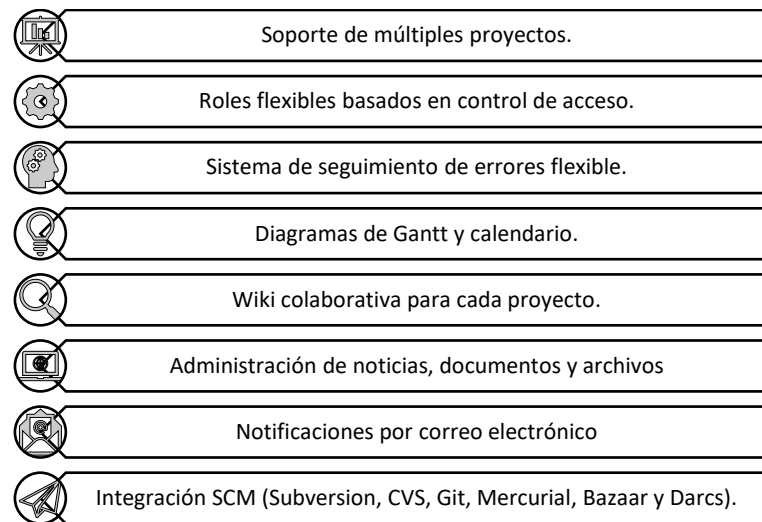
Es una herramienta para la gestión de proyectos y el seguimiento de errores, escrita usando el framework Ruby on Rails como base de datos, soporta tanto MySQL como PostgreSQL o SQLite. La principal ventaja que aporta, como gestor de proyectos, es toda la información asociada a un proyecto que contiene, esto permitirá controlar su ejecución, a través de una interfaz web que hace sencilla la gestión (Adriam, 2020).

Redmine incluye un calendario y los diagramas de Gantt para la representación visual de la línea del tiempo de los proyectos.

En la Figura 11 se puede apreciar las características de la herramienta REDMINE.

Figura 11

Características de la herramienta de computación en la nube REDMINE



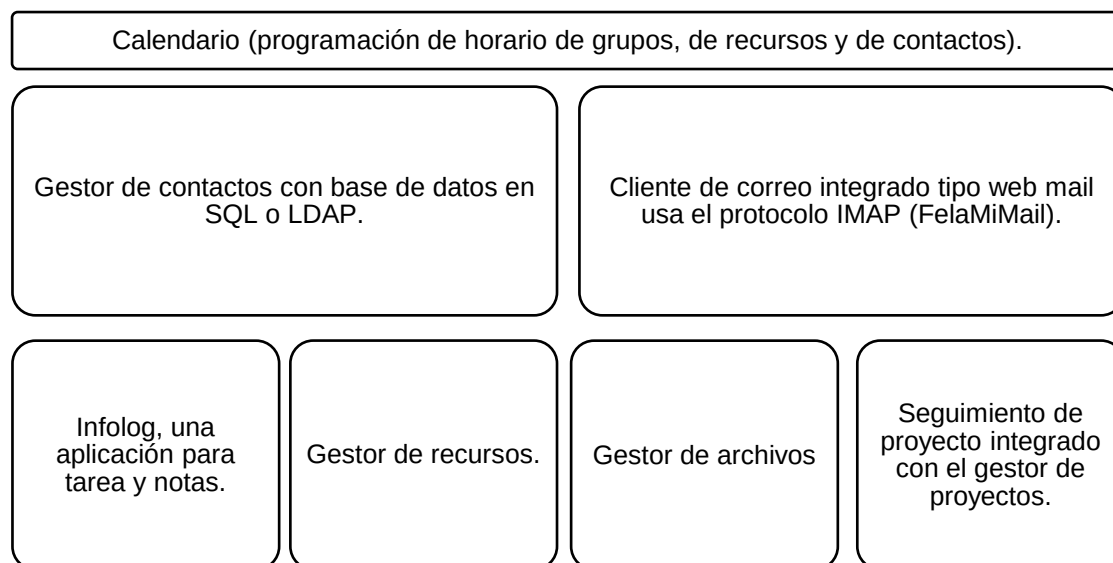
Nota. Características de la Herramienta REDMINE

2.5.8 Egroupware

Es una solución de trabajo en grupo vía *web*, de código abierto. Está escrita en PHP utilizando bases de datos, tales como LDAP, PostgreSQL, o MySQL. En la Figura 12 se puede apreciar las características de la herramienta antes mencionada.

Figura 12

Características de EGROUWARE



Nota. Características de la herramienta adaptado de EGROUWARE

2.5.8 Onlyoffice Formerly TeamLab Office

Es una plataforma de colaboración y gestión de proyectos bajo el paradigma SaaS (Software as a Service, o software como servicio), gratuita y de código abierto. Ofrece un calendario, gestión documental o un CRM como sistema simple y multifuncional para mejorar las relaciones con los clientes y aumentar las ventas. Puede ser una buena alternativa a Google Docs para pymes. Está pensado para el mundo profesional, pero su interfaz es muy sencilla y fácil de utilizar.

Tiene además interesantes aplicaciones como la integración de la serie OpenOffice en el navegador, permitiendo usar estos potentes ware recursos para editar textos, presentaciones, hojas de cálculo, entre otros archivos, sin salir del portal. Esta aplicación permite crear y administrar grupos de trabajo, gestionar proyectos y documentos, fijar objetivos y asignar tareas a usuarios definidos, además, trabaja con un sistema de notificaciones dirigido a los miembros del equipo para recordar las tareas asignadas.

2.5.9 Open Atrium

Es una herramienta que incluye, entre otras utilidades, una wiki, un gestor documental y un gestor de incidencias, Open Atrium es una herramienta open source, altamente personalizable, construida sobre Drupal, con una comunidad in crescendo y que, aunque su verdadero potencial se muestra con su utilización dentro de grupos bastante poblados, se adapta también a equipos pequeños. Construida sobre Apache, PHP, MySQL y un Core de Drupal, Open Atrium permite crear distintos grupos de trabajo y añadir usuarios a los mismos. Dentro de cada grupo se puede disponer de una serie muy interesante de características como calendario, gestor de incidencias, documentos organizados en libros, blogs, un escritorio muy bien estructurado e incluso una especie de muro similar a Facebook, muy útil para compartir enlaces, trucos y pensamientos.

2.5.10 Project Open

Es un conjunto de soluciones ERP open source para la gestión empresarial orientada a proyectos. Dentro del sistema project-open existen varios proyectos verticales:

Project-translation: solución para agencias de traducción.

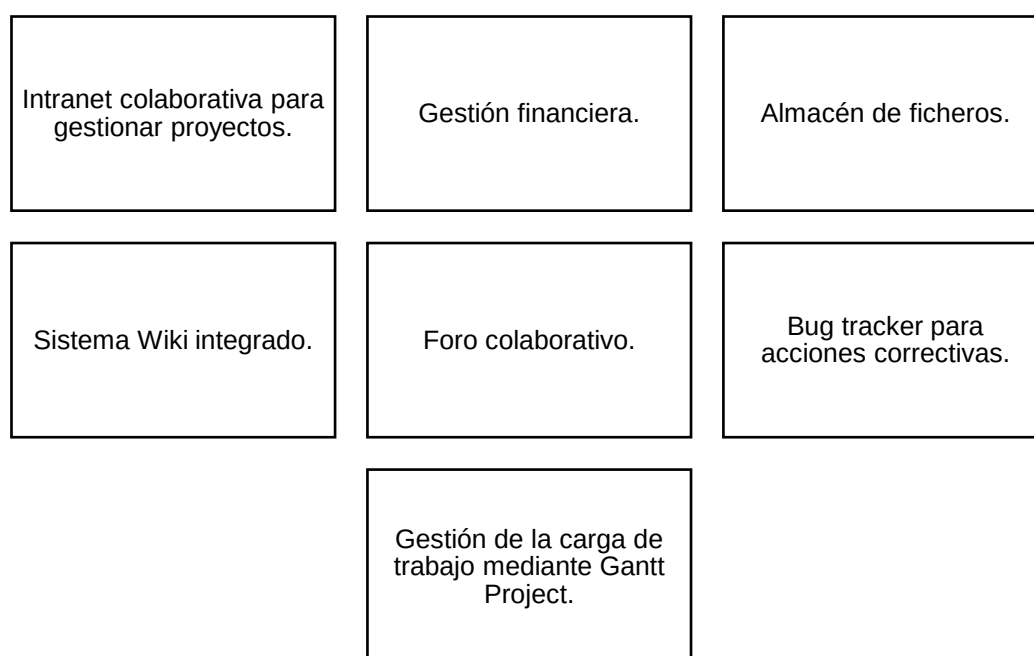
Project-consulting: para consultoras con intranet de plataforma colaborativa, gestión de proyectos, workflow y gestión del conocimiento.

Project-agency: solución para agencias de publicidad e Internet, que optimiza la gestión de los recursos internos de la empresa (Project Open Business Solutions S.L,2018).

En la Figura 13 se presentan algunas de sus principales características.

Figura 13

Características de la Herramienta Project Open



Nota. Características de la Herramienta Project Open adaptada de Project Open Business Solutions S.L(2018)

2.5.11 Jira

Es una aplicación web orientada al seguimiento de errores, incidentes y para la gestión operativa de proyectos. Jira también se utiliza en áreas no técnicas, para la administración de tareas (Atlassian,2022), para el desarrollo de software, como apoyo a la gestión de requisitos, seguimiento de estatus y más tarde para el seguimiento de errores. Entre las características de la Jira destacan:

1. Jira puede ser utilizada para la gestión y mejora de procesos, gracias a sus funciones para la organización de flujos de trabajo. Jira se basa en Java EE y funciona en varios sistemas operativos.
2. La flexible arquitectura de Jira permite al usuario crear ampliaciones específicas que pueden incluirse en la 'Jira extension library'.
3. Notificaciones vía email.
4. Gestión de incidencias.
5. Gestión de usuarios. Control de timing. Permite a cada miembro imputar el tiempo que ha dedicado a cada actividad y luego realizar informes (Atlassian,2022).

2.5.12 Stormboard

Es una fantástica herramienta 2.0 muy útil para realizar tormentas o lluvias de ideas entre varios usuarios; todo ello de forma *online* y con la posibilidad de añadir imágenes y otros elementos multimedia al tablero virtual donde se desarrolla la actividad. Con solo realizar el registro correspondiente se podrá crear tableros para las tormentas de ideas e invitar a los que vayan a participar añadiendo el correo electrónico. Es una herramienta práctica que tiene muchos usos educativos, especialmente para comenzar proyectos en los que es necesario intercambiar ideas, sugerencias, opiniones, entre otras, es útil también para organizar trabajos de clase.

2.5.13 Project2manage

Es una solución libre de gestión de proyectos organizados, que cualquier persona puede utilizar tanto en su vida cotidiana como profesional. Project2Manage es una gran herramienta para mantenerse organizado, tiene características que le permiten crear, gestionar y terminar los proyectos con diferentes actividades a diario

2.5.14 Teamwork.com

Es un software online sencillo de usar que facilita a los managers, empleados y clientes trabajar juntos de un modo más productivo. Entre las características de Teamwork.com destaca:

1. Rastrea errores y quejas de clientes.
2. Programas hitos, actividades y entregables.
3. Gestiona varios proyectos a la vez.
4. Distribuye recursos a un proyecto, solo cuando es necesario.
5. Mantiene todos los documentos asociados con un proyecto en un solo lugar, para facilitar la comunicación entre el equipo mientras trabajan en el proyecto.

2.5.15 Diigo

Este nombre es un acrónimo que se deriva del inglés "Digest of Internet Information, Groups and Other stuff". Diigo es un sistema de gestión de información personal basado en el concepto "nube", incluye marcadores web, bloc de notas post-it, archivo de imágenes, documentos y selección de textos destacados.

2.5.16. Editores de código online

AWS Cloud9 es un entorno de programación en la nube que permite escribir, ejecutar y depurar un código en el navegador. Como tal, incluye un editor de código, pero también un debugger y un acceso a terminal. Cloud9 viene ya con soporte para los lenguajes más populares, lo que permite empezar a programar y configurar . Y por supuesto, siendo un producto de Amazon, es muy fácil también escribir un código que utilice todos los servicios de Amazon. Tiene acceso directo a la instancia Amazon EC2 que aloja el entorno de desarrollo. Además, el editor permite la colaboración en tiempo real (Cabot, 2018).

Visual Studio Online. Microsoft ofrece el editor Monaco, una versión de Visual Studio Code capaz de ejecutarse dentro del contexto del navegador.

Codeanywhere. Es otro competidor fuerte, configurado para dar soporte a todo tipo de lenguajes de programación. Codeanywhere tiene varios entornos / proyectos preconfigurados. Por ejemplo, se puede programar en PHP, pero si el código PHP es para un plugin de WordPress, permite programar un proyecto WordPress con soporte específico para ese entorno.

Code Sandbox. Es un editor online especializado en el desarrollo web. Una funcionalidad muy útil de este editor es que cada vez que se guarda un proyecto se puede compartir la URL y cualquier usuario tiene acceso para modificarlo. Esta fue la motivación original que llevó a su creador a empezar el proyecto, además posee otras funcionalidades muy interesantes (como el soporte a NPM y la gestión de dependencias), pero la razón para incluirlo en esta lista fue su integración perfecta con GitHub. Basta con copiar la URL del proyecto en GitHub y pegarla en Code Sandbox para tener un nuevo entorno de programación ligado al proyecto, que permita modificarlo, y que además se mantenga actualizado con todos los cambios que otros hagan directamente en GitHub.

Las herramientas de CN previamente descritas fueron utilizadas en el marco de la presente tesis doctoral, ya que por sus características y naturaleza son adaptables a los entornos educativos y a los casos cuasi experimentales desarrollados para comprobar la eficiencia de las herramientas de CN, antes mencionadas, en el desarrollo de las habilidades de la programación.

2.6 Computación Afectiva

En la siguiente sección se presenta las conceptualizaciones referentes a la CA y su aplicabilidad en el medio educativo como base para el desarrollo de modelos pedagógicos que sean soportados, con sus principios y aplicaciones, en el contexto del desarrollo de las habilidades de programación.

2.6.1 Definiciones de Computación Afectiva (CA)

De acuerdo con Lixiang (2010), la emoción es un aspecto importante de la inteligencia humana y se ha demostrado que desempeña un papel fundamental en el proceso de toma de decisiones. Comprender las emociones es un aspecto relevante del desarrollo y crecimiento personal y, como tal, es una pieza clave para la emulación de la inteligencia humana (p.554). Además de ser importante para el avance de la IA, el procesamiento de

emociones también es necesario para la tarea estrechamente relacionada con la detección de polaridad (Rimal et al., 2009).

El aprendizaje colaborativo asistido por computadora (CSCL, por sus siglas en inglés) investiga el uso de tecnologías en contextos de aprendizaje colaborativo para promover la interacción y colaboración de los estudiantes, y maximizar su aprendizaje (Sawyer, 2008).

A continuación, presentamos algunas definiciones referentes a la CA.

El término Affective Computing, en español Computación Afectiva (CA), fue originalmente acuñado por Picard (1997): “área que estudia cómo los sistemas computacionales pueden reconocer y reaccionar a las emociones humanas, expresándolas por medio de una interfaz computacional”(p.6). Este campo de estudio considera dos características importantes en relación con las emociones humanas, la capacidad de reconocer el estado emocional de otro individuo y la experiencia social, en la que el individuo tiene la sensación de que su estado emocional es entendido por otras personas (Picard y Klein, 2002).

Desde que Picard (1997) propuso la CA, se incrementaron considerablemente las investigaciones enfocadas en crear tecnologías para monitorear y responder adecuadamente a los estados afectivos. A través de esas tecnologías, las computadoras pueden reconocer las emociones humanas de diferentes maneras. ¿Por qué es importante vincular la emoción humana con las computadoras? La respuesta está en la última investigación científica, que establece que las emociones juegan un papel esencial en la toma de decisiones, la percepción y el aprendizaje (Ben Ammar et al., 2020). Así mismo, CA se ha vuelto relevante debido al hecho de que la emoción juega un papel esencial en la toma de decisiones, la percepción y el aprendizaje (Wu et al., 2015).

De acuerdo con Calvo y D'Mello (2010), “la CA posibilita desarrollar sistemas computacionales de reconocimiento para los estados humanos, además estrecha el canal de

comunicación entre la computadora altamente emocional humana y la computadora emocional desafiada”(p.22).

Las técnicas de CA pueden observar las expresiones emocionales de los alumnos y luego proporcionar retroalimentación emocional para mejorar la motivación y la facilidad de uso y la interacción. La investigación de CA es un área multidisciplinar que tiene como objetivo explorar la experiencia humana efectiva con la tecnología, a través de la combinación de ingeniería y tecnología informática, con diferentes dominios de investigación como la psicología, la ciencia cognitiva, la neurociencia, la sociología, la educación, la psicofisiología, el diseño centrado en el valor y la ética (Tao y Tan, 2005).

De acuerdo con Banafa (2016), la CA es el estudio y desarrollo de sistemas y dispositivos que pueden reconocer, interpretar, procesar y estimular las emociones humanas. Es un campo interdisciplinario que abarca la ciencia, la psicología y la ciencia cognitiva de la computación. Su ámbito principal es el diseño de dispositivos de computación, cuyo propósito es exhibir las capacidades emocionales innatas que son capaces de estimular, de una manera conveniente, las emociones.

Tian et al. (2022) definen a la CA como:

Un campo naciente situado en la intersección de la inteligencia artificial con las ciencias sociales y del comportamiento, es decir la CA estudia cómo se perciben y expresan las emociones humanas, lo que luego da lugar al diseño de agentes y sistemas inteligentes que pueden imitar este comportamiento para mejorar su inteligencia o incorporar dicho conocimiento para comprender y comunicarse de manera efectiva con sus colaboradores humanos; esto llevado al contexto educativo permite comprender el estado emocional de los estudiantes y con ello trazar las estrategias pedagógicas necesarias para un adecuado proceso de enseñanza y aprendizaje soportada por TIC en especial las herramientas de CN (p.35).

2.6.2 Computación afectiva en la educación

Las emociones desempeñan un papel muy importante en la adquisición, procesamiento de información y en el aprendizaje de las personas; en consecuencia, dentro de la inteligencia artificial, en los últimos años, se ha visto un crecimiento vinculado a las investigaciones sobre el reconocimiento de las emociones como parte del área de la CA.

La CA busca comprender, interpretar y simular las emociones humanas. La idea es que las máquinas no solo realicen tareas de manera eficiente, sino que también puedan interactuar con los humanos de una manera más natural y sensible al contexto emocional. Esto incluye desde sistemas de reconocimiento facial, que pueden interpretar expresiones emocionales, hasta asistentes virtuales y chatbots que ajustan sus respuestas basadas en el tono emocional de la entrada de texto o voz.

Dentro de la literatura, en torno a la emoción y el aprendizaje, muchos autores determinan la relación directa que existe entre estos factores y cómo la emoción puede aumentar o inhibir el aprendizaje en los seres humanos (Graesser, 2007); por lo que, las emociones pueden mejorar la percepción y toma de decisiones que influyen en los propios mecanismos del pensamiento racional (D'Mello, 2014). En el mismo orden de ideas, se puede describir al aprendizaje como el acto de adquirir conocimiento, comportamientos, aptitudes y normas. De ahí la importancia de investigar el desarrollo de tecnología que brinde a los estudiantes un entorno cada vez más amigable y adaptado a las necesidades de cada uno de ellos (Hernández-Sánchez, 2015). En el mismo contexto, en la educación convencional, lo habitual es que el estudiante tenga que adaptarse a las condiciones que impone la escuela, el profesor e incluso aquello que pueden mencionar sus compañeros. Si realmente se busca que el aprendizaje sea efectivo, se deben considerar las capacidades y la disposición al aprendizaje de cada individuo, adaptando la manera de enseñar según el ritmo personal. En la actualidad, esto es posible gracias a la tecnología con el apoyo de herramientas como CN y CA.

El proceso cognitivo humano está compuesto por el pensamiento y el sentimiento (Piaget, 2005; Picard, 2000; Goleman, 1995; Vygotsky, 1994). Las

emociones son una parte muy importante en la vida del ser humano ya que permiten la interacción. Por medio de ellas se comunica lo que sentimos y pensamos, esta habilidad es esencial en el diario vivir. Por esto, es necesario que todo sistema artificial tenga en cuenta las emociones, para asegurar un buen sistema de comunicación hombre-máquina.

La tecnología avanza cada día. En la actualidad, se habla de ambientes inteligentes, computación ubicua y dispositivos inteligentes, elementos que están influenciando y cambiando la forma de vida de las personas (Pentland y Choudhury, 2000).

En el campo de la CA, la tecnología de reconocimiento facial ha despuntado en los últimos años gracias a los dispositivos móviles, el mejoramiento de los algoritmos matemáticos, el poder computacional y la interconexión que han permitido reconocer rostros en segundos (Lorenzo, 2016).

Las emociones son una parte fundamental del ser humano, ya que motivan las acciones y añaden significado y riqueza a toda experiencia humana. Tradicionalmente, las emociones han sido descartadas en la interacción humano-computadora. Sin embargo, investigaciones en psicología y tecnología sugieren un punto de vista diferente de la relación entre humanos, computadoras y emociones.

La CA ofrece muchas oportunidades en diferentes campos de aplicación, como por ejemplo en el entretenimiento, puede aparecer un programa que nos pone música de acuerdo con nuestro estado de ánimo; en actividades de oficina, se puede presentar un programa que no únicamente convierta en texto los mensajes que dictamos, sino que reconozca el tono en que lo decimos; en servicio al cliente, puede existir un programa que detecte el estado afectivo de los clientes al levantar un pedido o una queja, y de la misma forma se pueden desarrollar oportunidades en muchas otras áreas. Pero tal vez, una de las aplicaciones que más rápidamente podría desarrollarse, con una buena aceptación, es la del tutor afectivo (un tutor que adapte su instrucción, no solamente al desempeño del estudiante sino también a su estado afectivo, y, en consecuencia, responda adecuadamente en cada situación).

La investigación de la CA ha visto avances significativos recientemente y hace una transformación crítica de estudios exploratorios dirigidos a aplicaciones del mundo real, en el área de investigación emergente conocida como computación afectiva aplicada.

La CA tiene aplicaciones en muchas áreas: interacción humano-computadora, robótica, visión computacional, razonamiento, inteligencia artificial y otras. Se han desarrollado diversos programas con capacidades afectivas, por ejemplo, aplicaciones de medicina con alertas afectivas que ayuden a los usuarios a vencer sus estados emocionales negativos, además del desarrollo de programas de reconocimiento de lenguaje natural, y programas de educación, como la aplicación en sistemas de tutores inteligentes que se propone en este documento.

La importancia de la motivación y de la afectividad en el aprendizaje ha sido resaltada por psicólogos y pedagogos que expresan la manera en que las emociones afectan el aprendizaje (Goleman, 1995; Piaget, 2005; Vygotsky, 1994). Piaget expresa que es innegable que la afectividad tiene un papel acelerador o perturbador en el aprendizaje y expone que la mayoría de los estudiantes que tienen problemas con las materias de matemáticas fallan debido a un bloqueo afectivo (Piaget, 2005).

En el contexto educativo de la CA se intentan abordar los inconvenientes principales del aprendizaje en línea frente al aprendizaje presencial, tales como la capacidad de los docentes para adaptar la situación pedagógica al estado emocional del estudiante durante la enseñanza.

2.6.3 Modelos de CA

El primer modelo de emociones que se propuso para inteligencia artificial fue el Modelo Cognitivo de Emociones OCC (Ortony et al., 1988), que recibe su nombre de los apellidos de sus autores Andrew Ortony, Gerald L. Clore y Allan Collins, y fue presentado en su libro 'The Cognitive Structure of Emotions' (Ortony et al., 1988). Los autores de este modelo pensaron que una computadora no necesitaba tener emociones, pero que los

programas de inteligencia artificial sí requerían razonar acerca de las emociones de los usuarios para tomar buenas decisiones, especialmente aquellos programas de entendimiento de lenguaje natural, solución cooperativa de problemas y planeación.

Plataforma affective reasoner es uno de los trabajos más importantes en modelos de emociones (Elliott, 1992), que propone un prototipo de emociones para los sistemas multiagentes y sus interrelaciones sociales. Este trabajo consiste en una implementación y extensión del modelo OCC. La plataforma Affective Reasoner muestra cómo modelar las personalidades de los agentes y sus relaciones sociales. Affective Reasoner consta de emociones básicas y acciones inducidas por estas. Los agentes cuentan con un componente interpretativo y uno expresivo que dan a cada elemento su propia personalidad y percepción de otros agentes.

Modelo Cathexis. Un trabajo importante en modelos de emoción es la arquitectura Cathexis (Velásquez, 1997), dirigida a modelar la generación de emociones y su influencia en el comportamiento de agentes autónomos animados; este trabajo está fuertemente apoyado en trabajos de psicología, etología y neurobiología (p.12).

Modelo motivacional. Por el lado de la pedagogía, existen algunas propuestas importantes que plantean la importancia del estado afectivo en el aprendizaje. Por ejemplo, en De Vicente (2003) se propone un modelo motivacional basado en las características del estudiante (expresadas por él mismo), específicamente en su interacción y desempeño en el STI.

El modelo motivacional se divide en dos clases de variables: permanentes (características personales del estudiante) y temporales (representan el estado actual del estudiante).

Modelo pedagógico. En el MIT Media Lab se ha desarrollado el proyecto Affective Learning Companion (Kort et al., 2001a; Kort et al., 2001b), cuyo objetivo es construir un compañero computacional que ayude a los estudiantes a aprender. El compañero de

aprendizaje hace preguntas ocasionales al estudiante y observa su desempeño, buscando signos que ayuden a determinar su estado afectivo, como frustración, aburrimiento, curiosidad o diversión.

Modelado afectivo del estudiante. Antes de hablar del modelado afectivo del estudiante es necesario definir lo que significa o por lo menos establecer lo que se entiende con este término. El modelo del estudiante es una estructura que representa el estado actual del estudiante porque tiene conocimiento sobre él.

La estructura del modelo del estudiante contiene como mínimo información acerca del conocimiento del estudiante. Los sistemas de tutores inteligentes generan esta estructura y la utilizan para adaptar la instrucción a las necesidades particulares de cada usuario.

Modelado afectivo del estudiante en la plataforma affective reasoner. Dentro del comportamiento afectivo existen dos habilidades importantes: evaluar el estado afectivo del estudiante para responder adecuadamente y expresar reacciones afectivas. El objetivo general de la plataforma Affective Reasoner es que los agentes tengan estructuras para la evaluación del mundo, que puedan reutilizarse, de esta manera el agente tendrá conocimiento de cómo los estudiantes ven el mundo (cómo cree el agente que el estudiante se siente con respecto a las situaciones que se presentan, dadas las creencias que tiene sobre sus metas, principios y preferencias).

2.6.4 Consideraciones teóricas acerca de la CA en el proceso de enseñanza-aprendizaje de la Educación Superior

De acuerdo con Rodríguez-Arboleda et al. (2017):

Las TIC han ganado un profundo espacio en el contexto de enseñanza-aprendizaje que se enfrenta hoy a nivel global, no solamente para prever más conocimiento en los estudiantes, sino también en la transformación y forma para enseñar y aprender, con especial énfasis, en el uso interactivo de la información. Así mismo, la tecnología, como medio de enseñanza, no debe utilizarse únicamente como una herramienta de demostración o de desarrollo cognitivo, sino que debe aportar a la formación de

actitudes, emociones, sentimientos y comportamientos que contribuyan al desarrollo de la personalidad y habilidades del estudiante. En esto radica la importancia de abordar a la CA en los procesos de enseñanza y aprendizaje para la educación superior, en especial en las ciencias de la computación (p.172).

El campo de la CA involucra principalmente la convergencia de dos áreas de conocimiento: la psicología y la ingeniería. Los psicólogos aportan su tradición en la investigación de las emociones, sus discursos y sus métodos. Mientras que los ingenieros y científicos de la computación aportan herramientas que hacen uso de técnicas de machine learning, que luego serán aplicadas en diferentes contextos, en especial en el educativo.

En la Educación Superior, la CA se está empleando para el diseño de ambientes de aprendizaje ubicuos que proporcionen un aprendizaje significativo, contextual y experimental, desde la perspectiva de la educación tecnológica emocional. En especial dentro del área de las ciencias de la computación. El aporte de la CA en la Educación Superior se ve reflejado en los sistemas y entornos virtuales de enseñanza y aprendizaje, para que de este modo sean capaces de modificar su interacción con el estudiante de forma empática; pues está comprobado que el estado emocional del estudiante tiene un impacto significativo sobre sus capacidades cognitivas. De esta forma, sería posible conseguir una participación y un menor índice de abandono en escenarios de aprendizaje.

La CA en la Educación Superior, promueve el desarrollo de las competencias disciplinares, tecnológicas y emocionales, desde la creación de ambientes virtuales de aprendizaje para construir conocimientos, desarrollar competencias, y habilidades. Todo ello mediante la estimulación de los sistemas sensoriales que movilicen las emociones y los procesos cognitivos de manera interrelacionada (Rodríguez-Arboleda et al., 2017,p.173).

La CA promete transformar la educación superior al hacerla más receptiva a las necesidades emocionales y cognitivas de los estudiantes, lo que podría resultar en una experiencia de aprendizaje más efectiva y enriquecedora. Sin embargo, su éxito dependerá

de un enfoque cuidadoso y considerado que equilibre la innovación tecnológica con las consideraciones éticas y pedagógicas.

2.7 El valor formativo de la CN y la CA

Según Clough (2015) la CN:

Se ha convertido en un término cada vez más familiar; este modelo de cómputo permite *reducir costos*, ampliar el *intercambio de conocimientos* y mejorar tanto las normas como las prácticas educativas. La CN ya ha comenzado a solucionar los diversos retos que enfrentan escuelas, colegios, universidades, no solo disminuye problemas internos, también crea nuevas oportunidades en el ámbito formativo. Para el aprendizaje se requiere un entorno adecuado, con estudiantes motivados y educadores de calidad. El rápido progreso en tecnologías y servicios, que caracteriza la era actual, ha aumentado dramáticamente los datos e información disponibles en Internet. Existe la necesidad de un sistema que ayude permanentemente a los usuarios a acceder, editar y compartir información (p.5).

Las herramientas de CN como tecnologías informáticas parecen satisfacer tales requisitos en términos de suministro de datos sobre la red de forma permanente, teniendo en cuenta los factores de participación, privacidad y el coste del Servicio (Clough,2015).

Para el aprendizaje electrónico basado en la CN y CA, los maestros pueden realizar varias tareas, conferencias y muchas otras actividades sobre la nube, pero el valor agregado es la interacción. Los maestros obtienen mayores posibilidades de atraer la atención de un estudiante, haciendo que el proceso educativo sea divertido; pueden preparar pruebas en línea, crear mejores contenidos y comunicarse con los estudiantes en línea. Las tecnologías en la nube también ayudan a cubrir una audiencia más amplia de estudiantes y gestionar su proceso de estudio: evaluar las pruebas, tareas, proyectos y enviar la retroalimentación nunca ha sido tan fácil (Riddle, 2021).

De acuerdo con Casacuberta y Vallverdú (2010) “la habilidad de comunicarnos por medio de emociones es esencial en nuestro diario vivir, cualquier sistema artificial debería

considerar este aspecto para asegurar una buena comunicación hombre-máquina”(p.130). La necesidad de implementar las emociones en sistemas artificiales surgió a partir de querer mejorar los procesos cognitivos y la interacción con los seres humanos (Picard, 1997); trasladado al campo educacional, se ha evidenciado mejoras considerables en el desempeño académico de los estudiantes, concentración, aceptación del sistema y aprendizaje auto percibido.

Los sistemas educativos basados en CN Y CA deben permitir un mayor acceso y flexibilidad, además de generar una reducción de los costes, deben procurar la permanencia del estudiante y brindar mayores posibilidades para que puedan incorporarse al proceso de aprendizaje en cualquier momento de la vida. La CN proporciona un cambio importante, pero lo más importante es la transformación metodológica y didáctica que desarrolla el docente, además de los sistemas de comunicación y distribución de los recursos de aprendizaje. Por medio de las herramientas de CN los docentes y estudiantes tienen acceso a distintos contenidos, así el proceso de aprendizaje será nutrido por el material elaborado por los docentes y por lo que los estudiantes encuentren en la web, sobre alguna asignatura. En nuestro caso en el contexto de la programación en el área de la informática.

2.8 Aspectos éticos y Seguridad de Datos en CN

La CN ha transformado la forma en que almacenamos, procesamos y compartimos información, pero también plantea importantes retos éticos y de seguridad de los datos. A continuación, se detallan algunas consideraciones de los aspectos antes mencionados.

- Rendición de cuentas y responsabilidad: Los proveedores de nube y usuarios dan forma al panorama ético de la CN a través de sus interacciones. Ambas partes deben rendir cuentas del uso ético de la tecnología, asegurando que sus prácticas beneficien a la sociedad (Murphy & Rocchi, 2021).

Si bien los entornos en la nube ofrecen numerosos beneficios, también plantean importantes desafíos éticos relacionados con la privacidad de los datos, el control, el

cumplimiento legal y la seguridad. Las organizaciones en especial las educativas deben sopesar cuidadosamente estas consecuencias y aplicar medidas apropiadas para proteger los datos sensibles.

- Cumplimiento de la reglamentación: Las consideraciones éticas están estrechamente vinculadas a los marcos jurídicos de cada país (Soni & Dhurwe, 2024).
- Soberanía y jurisdicción de datos: Las cuestiones éticas también se derivan de la soberanía de los datos, cuando los datos almacenados en diferentes jurisdicciones pueden estar sujetos a diferentes requisitos legales, complicando el cumplimiento y la gestión ética de los datos(Soni & Dhurwe, 2024).

Dentro de este mismo orden de ideas, los desafíos en la seguridad de datos, que se presentan en entornos de CN destacan:

- Confidencialidad, Integridad y Disponibilidad
- Acceso no autorizado
- Cifrado de datos y transmisión segura
- Control de Acceso y Autenticación

Al integrar estas estrategias, las organizaciones pueden reducir significativamente los riesgos de acceso no autorizado a datos sensibles almacenados en entornos en la nube, asegurando una experiencia segura y confiable de CN en especial en el contexto educativo.

Abordar los desafíos éticos y de seguridad de los datos en la CN requiere un enfoque multifacético que incluya medidas de seguridad sólidas, el cumplimiento de los marcos legales y una sólida base ética. Las evaluaciones periódicas, las tecnologías avanzadas y la colaboración efectiva entre los interesados son fundamentales para garantizar prácticas de CN seguras y éticas.

Capítulo tres: La dimensión pedagógica

3.1. La lectura pedagógica del uso educativo de la tecnología

Un gran desafío que tiene el mundo actual es innovar en educación (Robinson, 2006), el reto se hace evidente en la propuesta de nuevos modelos pedagógicos que sean soportados por la tecnología. Por ello, en la presente sección se propone una lectura pedagógica del uso educativo de la tecnología. Es necesario manifestar que la pedagogía, tal cual la entendemos, es la disciplina que especifica los objetivos y las técnicas de enseñanza-aprendizaje, precisando los objetivos y los medios para lograr los fines de la educación (Salcedo, 2011); así mismo, la tecnología trata de efectuar una síntesis entre la verdad y la eficiencia, entre la ciencia y la técnica, entre los sistemas de reglas y los sistemas de leyes.

La tecnología se funda en los conocimientos teórico-empíricos de la ciencia y los usa. También utiliza la técnica, que es un sistema de reglas de operación eficaz para modificar la realidad (Aguilera & Rina; 2013,p.82).

De acuerdo con Salcedo (2011) la pedagogía actual:

Debe coordinar y ofrecer visiones sobre estas nuevas formas de representación educativa. No basta con una respuesta mecánica, basada únicamente en las posibilidades tecnológicas, sino que también se debe considerar un enfoque interdisciplinario en el proceso formativo para presentar las bases multitudinarias que sustentan el cambio educativo; dentro de estos cambios destaca la Pedagogía Digital, que se está convirtiendo en una parte integral de las ciencias pedagógicas (p115.).

Esto viene dictado por el hecho de que los países avanzados del mundo se están moviendo hacia el formato de la economía digital (Lizunkov et al., 2021; Mau, 2020) y a las diversas manifestaciones humanas que en la praxis se llevan a cabo a diario.

En este contexto, la economía digital se ha convertido en un fenómeno social importante, con un impacto significativo en muchas esferas de la vida. La transición digital es

esencial para aprovechar al máximo las ventajas de la economía digital y para mantenerse competitivo en un mundo cada vez más conectado. Esto incluye la implementación de tecnologías avanzadas en áreas como la educación, la atención médica, la gestión empresarial y la investigación científica.

El ámbito pedagógico juega un papel vital en la formación profesional en todas las áreas. Por eso, se imponen requisitos especiales a la calidad del funcionamiento y al nivel de desarrollo de la pedagogía digital (Peshkova & Samarina, 2018,p.25)

Considerando las definiciones previas, contextualizaremos a la Pedagogía Digital como una rama de la ciencia pedagógica que revela la esencia y regularidades de la educación digital además del papel de los procesos educativos “digitalizados” en el crecimiento personal, el cual desarrolla formas prácticas y medios para mejorar su eficacia (Ilaltdinova et al., 2019). En este contexto, Toktarova & Semenova (2019) enfatizan que el término “pedagogía digital” es bastante voluminoso en su significado funcional y contenido informativo, en él se distinguen algunos componentes principales:

- Basado en el contenido: crear un producto educativo digital, con nuevas oportunidades de aprendizaje y comunicación cognitiva de los sujetos del proceso educativo. En nuestro contexto, un modelo pedagógico basado en CN y CA contribuye a la mejora de las habilidades de programación.
- Ambiental: transferencia del componente de contenido y comunicación al entorno educativo digital y transformación del proceso de enseñanza y aprendizaje dentro del marco del modelo “profesor – entorno educativo digital – alumno”.
- Tecnológicos: formularios (sincrónicos, asincrónicos), métodos (activos, interactivos, etc.), herramientas (ordenadores, portátiles, teléfonos móviles, recursos educativos electrónicos, etc.) y la docencia técnica (tecnologías multimedia, tecnologías en la nube, etc.).
- Basado en competencias: formación y mejora de las competencias digitales de los docentes para proporcionar un aprendizaje interactivo y significativo para los estudiantes (Ilaltdinova et al.,2019).

“ En este mismo orden de ideas, la educación tiende a extraer su concepción del cambio tecnológico del entorno cultural prevaleciente ” (Plowman & McPake, 2013,p.28). La promesa de que la educación digital revolucionará la enseñanza y el aprendizaje, a través de la amplia disponibilidad de recursos digitales o experiencias de aprendizaje virtual radicalmente reestructuradas, a menudo pasa sin comentarios sobre los supuestos sociales, éticos y epistémicos problemáticos que sustentan tales cambios (Lewin & Lundie,2016, p.236).

Díaz (2012) sostiene que la formación universitaria basada en las TIC [...] se engloba en tres dimensiones diferentes con funciones propias: dimensión organizativa, dimensión tecnológica y dimensión pedagógica. Con relación a este aspecto, no todas las posibilidades que las TIC ofrecen en el campo educativo son aprovechadas por igual en todas las instituciones universitarias, ni las utilizan con la misma intensidad. Díaz (2012) también plantea que la dimensión pedagógica se relaciona con las propiedades del proceso de enseñanza y aprendizaje mediadas por las tecnologías. Esta se divide en tres subdimensiones que se derivan en funciones propias: metodología utilizada, sistema de comunicación y materiales didácticos (p.50).

Las TIC, desde esta perspectiva pedagógica, son un catalizador para la transformación educativa y para el desarrollo de nuevos roles tanto para los estudiantes como para los docentes, al ser crucial en el afrontamiento de los desafíos que plantean la sociedad y el sistema en cualquier nivel educativo. Desde lo metodológico hasta la elaboración de contenidos, métodos organizativos, colaboración y de evaluación.

Al mismo tiempo, las TIC han permitido el desarrollo profesional docente en comunidades de aprendizaje y el aprendizaje colaborativo no presencial, en contextos de enseñanza y aprendizaje dentro de la educación básica y superior. Expresado lo anterior se puede diferir que la lectura pedagógica del uso educativo de la tecnología se refiere a cómo los docentes y los diseñadores de programas educativos utilizan herramientas tecnológicas

para mejorar el aprendizaje de los estudiantes. Esto incluye el uso de software educativo, aplicaciones móviles, juegos educativos, plataformas en línea y otros recursos tecnológicos para desarrollar habilidades y conocimientos. La lectura pedagógica también implica considerar cómo las herramientas tecnológicas se utilizan para apoyar los objetivos y metas educativos, y cómo se evalúa el impacto de su uso en el aprendizaje de los estudiantes.

3.2 Análisis pedagógico de la educación con tecnología en pandemia

Ilaltdinova et al. (2019) manifiestan:

Que la pandemia mundial fue, por supuesto, no solo una grave emergencia de salud pública, sino también una emergencia política, económica y social. En un área clave como es la educación, creemos que el aprendizaje, los medios y la tecnología hicieron una contribución más directa al conocimiento y la práctica durante la pandemia de COVID-19, el cambio a formatos de educación en línea y digitales, y el surgimiento de formas "remotas" de enseñanza y aprendizaje. En este momento de política pandémica, donde se libran concursos en múltiples escalas y niveles sobre las formas de manejar y resolver la crisis, la educación a distancia se ha convertido en un motivo de preocupación generalizado para las autoridades políticas, las empresas educativas, las organizaciones benéficas, los maestros, los padres y los estudiantes (p.35).

La educación fue un asunto de emergencia, en el que las tecnologías educativas se posicionaron como un servicio de primera línea. La pandemia de COVID-19 afectó la educación en todos los niveles de varias maneras. Las instituciones y los formadores de docentes tuvieron que responder rápidamente a una transición inesperada y forzada de la enseñanza presencial a la remota. También tuvieron que crear entornos de aprendizaje para los estudiantes de magisterio que se preparaban a la luz de los requisitos de los programas de formación docente y las condiciones en las que tanto las universidades como las escuelas tenían que operar.

De acuerdo con Ilaltdinova et al. (2019):

El brote de coronavirus (COVID-19) causó varios niveles de interrupciones en la educación a nivel mundial, de ello, Brailovsky (2018) manifiesta que es importante establecer que el uso de las nuevas tecnologías computacionales en la enseñanza actual permite a los estudiantes explorar, inferir, hacer conjeturas, justificar, contrastar argumentos y así construir su propio conocimiento. La pandemia ofreció valiosas lecciones sobre la capacidad de adaptación del sistema educativo y destacó la importancia de la flexibilidad curricular y la preparación tecnológica. También subrayó la necesidad de estrategias de inclusión digital para garantizar que todos los estudiantes tengan las mismas oportunidades de aprendizaje. En conclusión, la pandemia no solo transformó la manera en que se imparte educación, sino que también puso de relieve la importancia de la tecnología en el ámbito educativo, al tiempo que expuso las desigualdades existentes y generó un impulso hacia métodos de enseñanza más flexibles y accesibles (P.36).

Así mismo, Floridi (2015) expresa:

Las TIC no son meras herramientas, sino fuerzas sociales que afectan cada vez más nuestra autoconcepción (quiénes somos), nuestras interacciones mutuas (cómo nos socializamos), nuestra concepción de la realidad (nuestra metafísica), y nuestras interacciones con la realidad (nuestra agencia). En cada caso, las TIC tienen un enorme significado ético, legal y político, que incide en el proceso de enseñanza y aprendizaje, más aún en tiempos de pandemia (p.15).

La enseñanza y el aprendizaje durante una pandemia son importantes, pero la salud, la seguridad y el bienestar deben ser lo primero en todo momento, y las escuelas y los maestros deben operar con una comprensión de las complejidades de la vida en el hogar y/o el estado mental, emocional y tensión física a la que se enfrentan sus comunidades.

Suponiendo que los padres, estudiantes y maestros estén listos para la distancia aprendizaje, la pregunta es: ¿Cómo se debe entregar? ¿Debería el aprendizaje en línea, ser parte del paquete de aprendizaje a distancia? Muchos elementos necesitan ser ponderados. Desigualdades en el aprendizaje en línea incluyen el acceso a la infraestructura, la disponibilidad de software, hardware, ancho de banda, costo y capacitación del personal, por nombrar algunos.

Otro desafío para pensar es que la educación en línea puede no ser apropiada para jóvenes estudiantes, especialmente para aquellos que tienen una mínima supervisión mientras están en casa. Pueden estar distraídos, preocupados, perder seguimiento de sus horarios o puede ser negativamente impactados por interacciones sociales limitadas o comentarios realizados por sus maestros en fechas anteriores a las que ellos ingresan. En algunos casos, el tiempo que deberían destinarlo a sus estudios pueden utilizarlo para pasar sus días en la naturaleza, cocinando, haciendo tareas domésticas y en juegos significativos.

Un desafío adicional es la cultura tecnológica entre los docentes y el acceso a la tecnología, se necesita un adecuado desarrollo profesional para contar con maestros listos que diseñen el aprendizaje a distancia y gestionen el componente tecnológico del aprendizaje en línea.

El esfuerzo por posicionar las pedagogías pandémicas como una oportunidad experimental natural, para que la ciencia de datos educativos "demuestre" los beneficios de la enseñanza digital ejemplifica las formas en que la "datificación", se ha presentado como una fortaleza transformadora en la educación en los últimos años. El uso de la tecnología en la educación se ha vuelto esencial debido a la pandemia de COVID-19, ya que permite continuar con el proceso educativo a pesar de las restricciones de distanciamiento social.

Desde un punto de vista pedagógico, se ha demostrado que la educación en línea y el uso de plataformas virtuales son una herramienta efectiva para el aprendizaje a distancia. Sin embargo, también se han presentado desafíos, como la falta de acceso a la tecnología y la falta de capacitación en su uso, especialmente en comunidades con bajos recursos económicos.

3.3 Definiciones de modelo pedagógico

Avendaño (2013) manifiesta que “la educación ha variado notablemente y con ella los modelos educativos y pedagógicos que, gracias a los avances científicos, hacen posible construir nuevas y mejores formas de enseñanza. A continuación, se presentan algunas definiciones referentes al modelo pedagógico ” (p.115).

Un modelo pedagógico es el marco teórico del cual se desprenden lineamientos para organizar los fines educativos, definir, secuenciar y jerarquizar los contenidos. También precisan las relaciones entre estudiantes, saberes y docentes. Determinan la forma en que se concibe la evaluación (De Zubiría, 2003), el autor manifiesta que los modelos pedagógicos y su comprensión son importantes en la medida en que posibilitan “reconocer las huellas o rastros que permiten reconstruir aspectos de la vida humana y que sirven de base para la reflexión y la investigación”.

Un modelo pedagógico es una amplia representación de un fenómeno y a partir de él se pueden abrir espacios para la reflexión dentro del quehacer pedagógico (Zubiría, 2003). Los modelos pedagógicos pueden ser de dos tipos: hetero-estructurantes y auto-estructurantes. Dentro del primer tipo se clasifica el modelo conductista y tradicional, y dentro del segundo, el aprendizaje significativo, el constructivismo o la Experiencia de Aprendizaje Mediado (EAM) (De Zubiría, 2007).

Un modelo pedagógico puede ser definido como representaciones “particulares de interrelación entre los parámetros pedagógicos” (Flórez, 1999,p.226), es decir, determina cómo son las relaciones entre los elementos que se involucran en el proceso de enseñanza-aprendizaje: el maestro, el educando, el objeto de estudio, el entorno, etc. El objetivo de un modelo pedagógico es hacer efectivo el traspaso del conocimiento en contextos socioculturales específicos, los cuales se interrelacionan en una esfera de complejidad. Es

urgente el diseño y puesta en práctica de modelos que aíslen las materias y asignaturas, y promuevan procesos de formación.

“Un modelo pedagógico plantea de forma integral un determinado fenómeno; desde el punto de vista teórico-práctico, ofrece un marco de referencia para entender implicaciones, alcances, limitaciones y debilidades paradigmáticas que se dan para explicarlo ” (Gimeno & Pérez, 1992,p. 448).

Son representaciones, pero dicha definición no permite identificar todos los elementos epistémicos que componen el concepto de modelo pedagógico. Un modelo constituye un "pilar meta-teórico sobre el que se edifican las ciencias naturales" (Galagovsky & Adúriz-Bravo, 2001). Ahondando en el tema de los modelos pedagógicos, se trata de constructos en los que se establece "un proceso o entidad del cual se supone su existencia aun cuando no sea directamente observable o medible" (Coll et al., 1994).

Un modelo pedagógico plantea de forma integral un determinado fenómeno, desde el punto de vista teórico-práctico ofrece un marco de referencia para entender implicaciones, alcances, limitaciones y debilidades paradigmáticas que se dan para explicarlo (Gimeno & Pérez, 1992). La sociedad de la información se ha convertido en la sociedad del conocimiento y del aprendizaje. Los sistemas educativos no quedaron alejados de estos avances, enseñar desde la virtualidad era un nuevo reto, esto provocó disímiles proyectos destinados a este fin. De acuerdo con García (2017) el uso de las TIC:

Promueve diversos cambios en el comportamiento de las personas o grupos sociales para que sean considerados miembros de los entornos digitales; sin embargo, para adquirir estas ventajas se requiere desarrollar habilidades importantes, tales como el diseño de una identidad digital que sea coherente con el uso de los diversos dispositivos y plataformas, con la finalidad de lograr una reputación en línea que sea favorable y lo consolide como fuente fiable. Esto nos lleva a pensar en nuevas propuestas de modelos pedagógicos con tecnología; por ello, la incorporación de las herramientas de CN para masificar su uso en los procesos de desarrollo de las habilidades de programación requiere un replanteo de los modelos pedagógicos y

paradigmas educativos existentes hasta el momento; el desarrollo de habilidades específicas en cada estudiante para facilitar su accesibilidad en este medio potenciando el proceso de enseñanza–aprendizaje(p.73).

Algunos ejemplos de modelos pedagógicos son el enfoque tradicional, el enfoque centrado en el alumno y el enfoque basado en proyectos. Esto hace necesario diseñar un modelo pedagógico para el mejoramiento de las habilidades de programación en los profesionales que se forman en el área de ciencias de la computación y afines, ese es el objetivo y propósito de la presente investigación.

3.4. Tipos de modelos pedagógicos

Avendaño y Parada (2011) “ expresan que la educación es un proceso sociocultural en el cual se reproducen las formas, costumbres, conocimientos, normas, conductas y paradigmas de una determinada sociedad en un tiempo determinado ” (p.399). En este marco, aparecen modelos y teorías para resolver los problemas de la educación respecto del entorno de los sujetos de aprendizaje; a pesar de las transformaciones profundas de las últimas décadas, en todas las esferas, la educación y la pedagogía han quedado rezagados a viejos modelos educativos. A la hora de construir o elegir un modelo pedagógico se debe tener en cuenta que responda a cuestiones fundamentales.

El modelo pedagógico no es solamente para materializar los objetivos académicos de los estudiantes, sino que debe permear e irradiar todas las acciones y prácticas de la propuesta educativa frente a todos sus miembros: estudiantes, docentes, padres, administrativos, comunidad local, etc. Entre los modelos pedagógicos se encuentran el tradicional, experimental-romántico, cognitivista desarrollista. En la tabla 3 se presentan las características más relevantes de cada uno.

Tabla 3
Características de modelos pedagógicos

Modelo	Descripción
Tradicional	En este modelo se concibe al estudiante como un ser pasivo, es decir, un receptor del conocimiento y objeto de la acción del maestro. El conocimiento se considera como algo dado y determinado por un sabedor exclusivo que es la teoría y/o el docente (Flórez, 2001).
Experiencial o activista	En este modelo el eje del desarrollo es lo que procede del interior del niño. El estudiante es el centro del proceso educativo, la interacción con el conocimiento y la experiencia son la fuente primaria para la activación de esquemas mentales y la construcción de conocimiento. Este modelo rescata al estudiante en su rol de conductor activo de sus propios aprendizajes, y a la realidad como el punto de partida y objetivo del aprendizaje.
Cognitivista desarrollista	En este modelo surgen diferentes tendencias o corrientes, las primeras conciben que la meta del proceso educativo consiste en que los individuos accedan, progresiva y secuencialmente, a la etapa superior de desarrollo intelectual; para esto, el maestro debe crear un ambiente que estimule la estructura cognitiva superior del niño o niña. No importa el contenido, lo que importa es el desarrollo cognitivo del sujeto (Flórez, 2001).
Conductista	Los estudios de Skinner y Pavlov sobre los aprendizajes son las bases fundamentales del modelo conductista que busca generar los medios para llegar al comportamiento esperado y verificar su obtención; sin embargo, nada garantiza que el comportamiento externo se corresponda con el mental (Flórez, 2001).
Social-cognitivo	Este modelo combina el trabajo productivo y la educación como una interacción que permite a los estudiantes el desarrollo del espíritu colectivo, el conocimiento científico-técnico y el fundamento de las prácticas sociales.

Nota. Características de modelos pedagógicos

3.5. Modelos Pedagógicos con uso de tecnología

Las nuevas tecnologías representan un reto para el profesorado universitario. El uso generalizado de Internet y de medios audiovisuales integrados cambiaron el entorno docente tradicional, las tecnologías generan cambios estructurales en la organización y desarrollo de los cursos, mediante la incorporación de nuevos modelos de aprendizaje. Las TIC han transformado la educación superior, modificando la manera de aprender y de enseñar, su expansión ha generado nuevas maneras de entender la forma cómo las personas se sitúan en la sociedad, en relación consigo mismas y con los demás (Vaillant y García, 2012, p.242). Esto ha dado lugar al surgimiento de nuevos estilos de aprendizaje, donde se requiere que el docente se transforme e innove a sí mismo (Torres y Alcántar, 2011).

El desarrollo de las TIC nos ha llevado a una sociedad digital, a un conocimiento y unas redes que cambiaron considerablemente la manera de comportarnos y vivir en sociedad a través de la cibercultura. Lemos y Cunha (2003) definen el modelo sociocultural como algo que emerge de la relación simbiótica entre la sociedad, la cultura y las nuevas tecnologías.

El uso de las TIC “ha centrado la atención de educadores, investigadores y gestores en Educación Superior, preferentemente en los últimos veinticinco años” (Bartolomé y Gallego, 2019, p.75). Es necesario el uso de plataformas digitales en el proceso de enseñanza y aprendizaje, que se basen en modelos pedagógicos sólidos, para generar un adecuado ambiente de aprendizaje. La presencia de las TIC en la educación, la introducción de recursos y herramientas tecnológicas ha crecido en los últimos años y se aprecia una manera diferente de concebir la enseñanza y el aprendizaje con un mayor énfasis en el aprendizaje autónomo y significativo sustentado en la tecnología.

El aprendizaje para el uso de las TIC precisa de una mayor efectividad y aunque se han desarrollado modalidades como el e-learning (aprendizaje por medio de la Internet) y el blended learning (aprendizaje matizado), aún es necesario continuar abordando las teorías

pedagógicas que lo sustentan y que garantizan un cambio didáctico y metodológico en su implementación (Montoya- Acosta et al., 2017, p.250).

De acuerdo con Sosa (2015), los beneficios del b-learning se han sustentado desde la teoría del constructivismo. Los sistemas b-learning son fácilmente adaptables al modelo basado en la solución de problemas que deriva en el constructivismo, porque se enfoca en la obtención del aprendizaje por iniciativa del alumno, motivándolo a la autonomía para propiciar el desarrollo de destrezas (p.114).

De acuerdo con Badillo (2011), el e-learning es una modalidad de formación basada fuertemente en las tecnologías de la información y comunicación, llamada también “formación virtual”, “formación online o en línea”. La formación en e-learning se basa generalmente en el uso de una aplicación web llamada “plataforma virtual de aprendizaje” para entregar los recursos a los participantes, quienes a través de equipos de cómputo y/o dispositivos móviles, ingresan ahí para visualizar los contenidos y completar las actividades antes diseñadas (p.20). En la tabla 4 se presentan las características de e- learning.

Tabla 4
Características del E-learning

Se promueve la interacción directa con los contenidos.
Uso intensivo de las TIC.
Usuarios dispersos geográfica o temporalmente.
Separación física entre instructor/docente y estudiante.
El estudiante como centro de la formación.
Autogestión del aprendizaje.
Tutorización.
Comunicación bidireccional entre estudiantes y formadores, de forma síncrona o asíncrona.
Registro de actividades realizadas.
Reducción de costos de formación / capacitación.

Flexibilidad para modificar los contenidos.

Nota. Características del E-learning

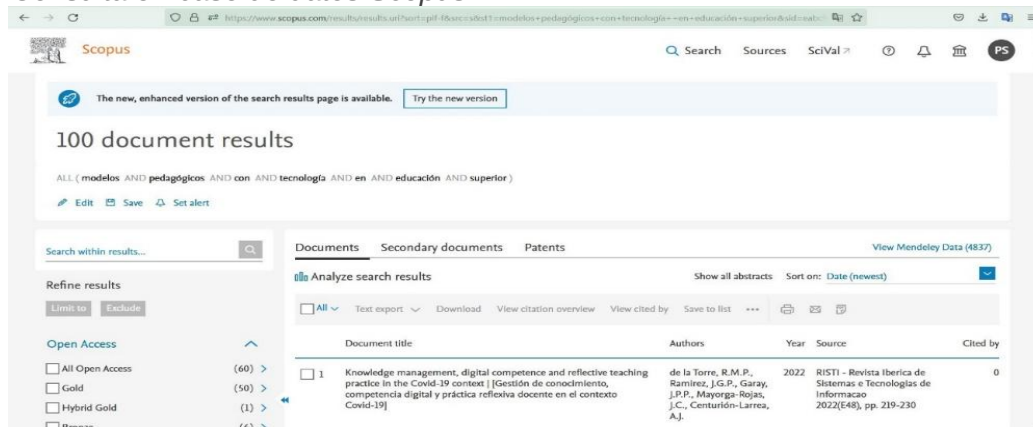
El e-learning, o aprendizaje electrónico, tiene varias ventajas. En primer lugar, permite el acceso a la educación a quienes de otra manera podrían tener dificultades para asistir a clases, como, por ejemplo, personas con discapacidades, personas que viven en áreas rurales o personas con responsabilidades familiares que no les permiten asistir a clases presenciales. Además, el e-learning permite a los estudiantes trabajar a su propio ritmo y en su propio horario, lo que puede ser especialmente útil para adultos que estudian mientras trabajan. También puede ser más eficiente y económico que la educación tradicional, ya que los materiales de estudio en línea suelen ser más accesibles y los costos de infraestructura son menores.

3.6 Estado del arte sobre los modelos pedagógicos con tecnología en educación superior

En el presente apartado se presenta la revisión sistemática de literatura basada en la propuesta por Kitchenham et al. (2002) el protocolo desarrollado se enfocó en responder la pregunta: ¿Qué modelos pedagógicos con tecnología se desarrollan en educación superior?, en el rango de los años 2012 al 2022.

Las búsquedas se hicieron en las bases de datos Scopus, WOS, IEE, ACM y Scielo. En la base de datos Scopus se obtuvieron 100 artículos relacionados con la pregunta de investigación. En la Figura 14 se pueden apreciar los resultados obtenidos.

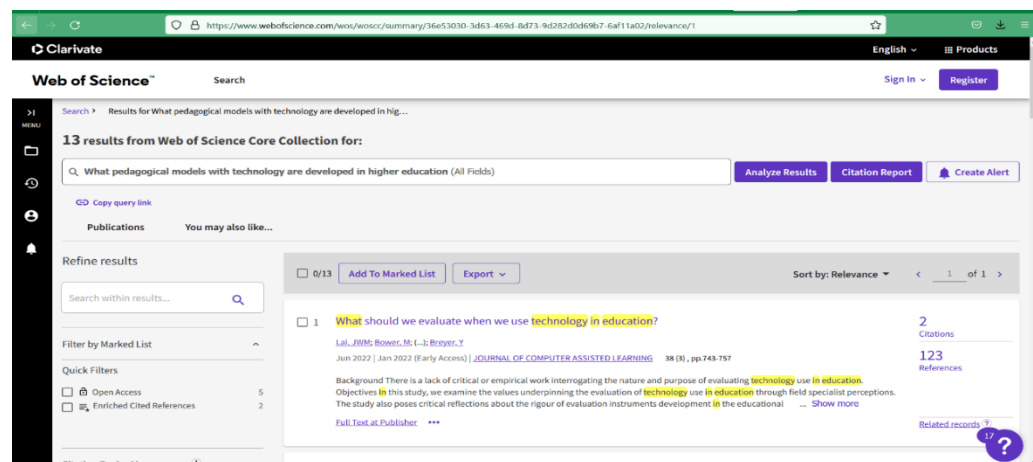
Figura 14
Consulta en base de datos Scopus



Nota. Datos encontrados en base de datos Scopus

En la base de datos WOS se encontraron 13 artículos con la pregunta de investigación propuesta (traducida al inglés) Ver Figura 15.

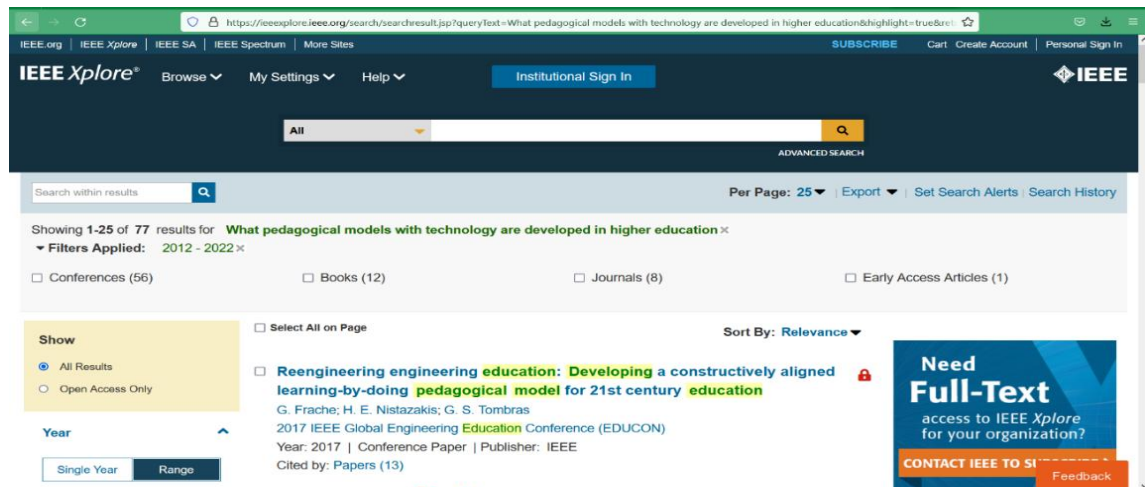
Figura 15
Consulta de base de datos WOS



Nota. Consulta en base datos WOS

En la base de datos IEEE se encontraron 77 recursos, entre ellos: 56 artículos en congreso, 12 libros, 8 artículos y 1 artículo acceso pronto con la pregunta de investigación propuesta (traducida al inglés). Ver Figura 16.

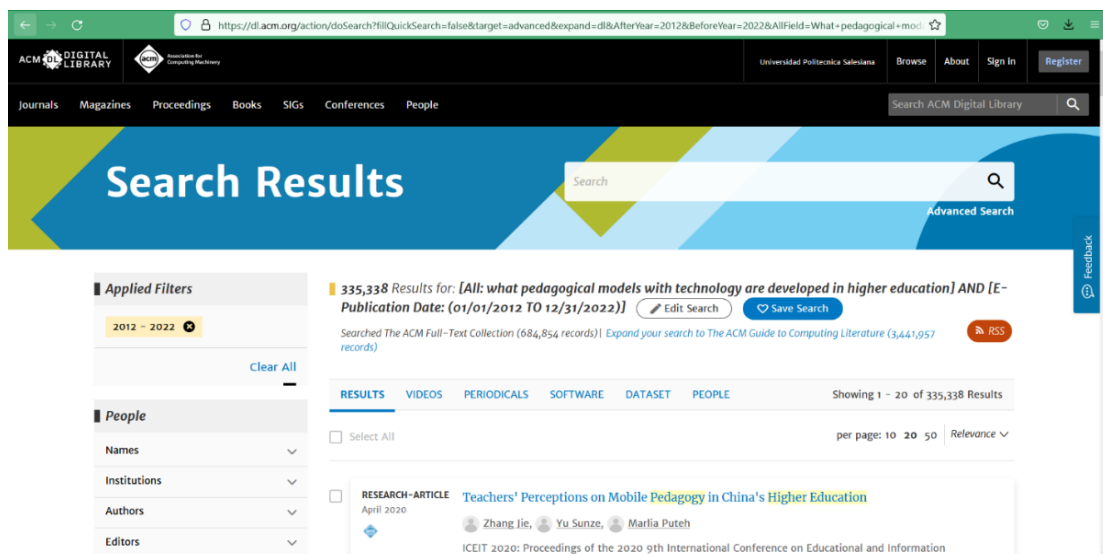
Figura 16
Consulta de base de Datos IEEE



Nota. Consulta en la base de datos IEEE Xplore

En la base de datos ACM se encontraron 335,338 recursos. Ver Figura 17.

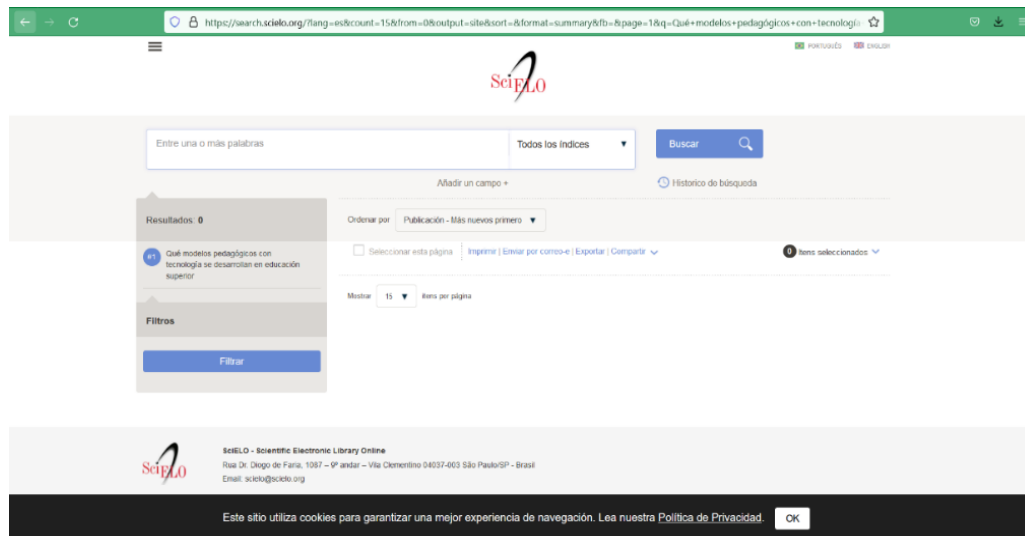
Figura 17
Consulta en base de datos ACM



Nota. Consulta de base datos ACM

En la base datos Scielo no se encontró ningún recurso con la pregunta de investigación propuesta; ver Figura 18.

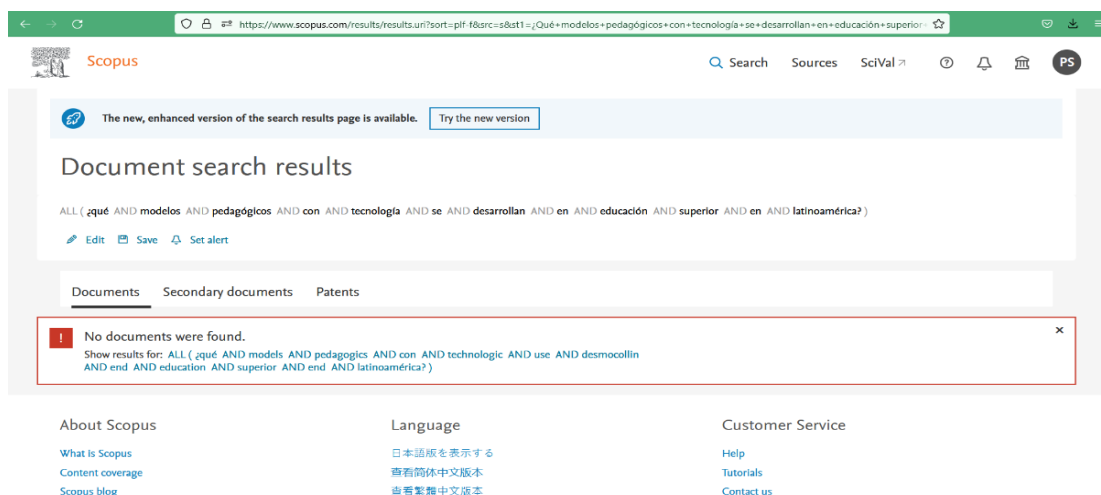
Figura 18
Consulta Scielo



Se realizó la misma pregunta, pero enfocada al contexto latinoamericano. ¿Qué modelos pedagógicos con tecnología se desarrollan en educación superior en Latinoamérica? en un rango del 2012 al 2022.

En la base de datos Scopus no se encontraron resultados a la pregunta de investigación en español (Ver Figura 19).

Figura 19
Consulta de base datos Scopus

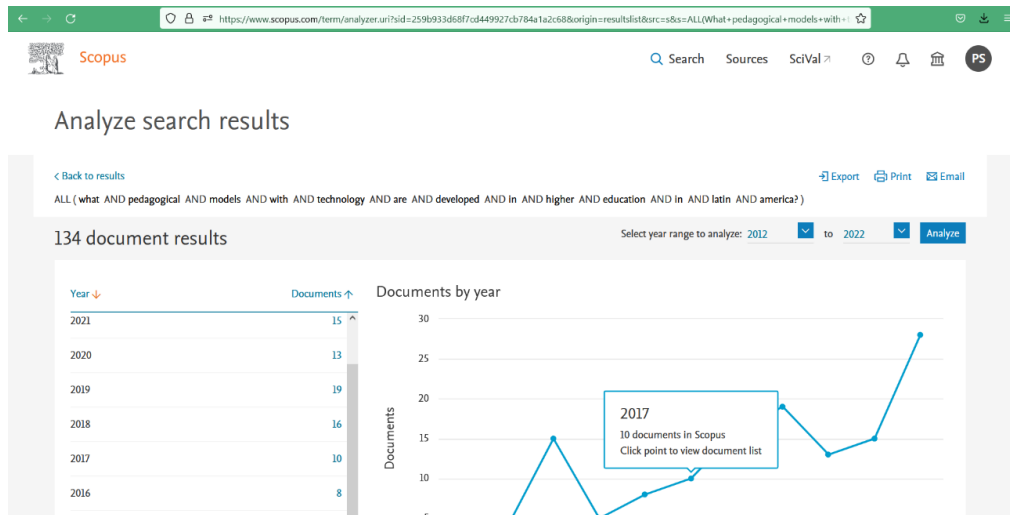


Nota. Consulta en la base de datos Scopus

Se realizó una segunda búsqueda en idioma inglés:

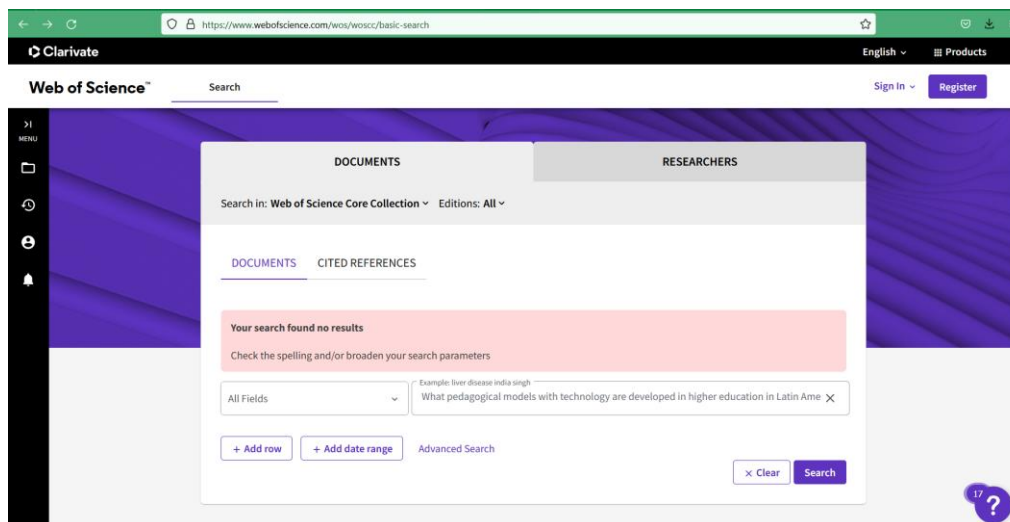
What pedagogical models with technology are developed in higher education in Latin América? Obteniendo 134 resultados (Ver Figura 20).

Figura 20
Consulta en Base de Datos Scopus



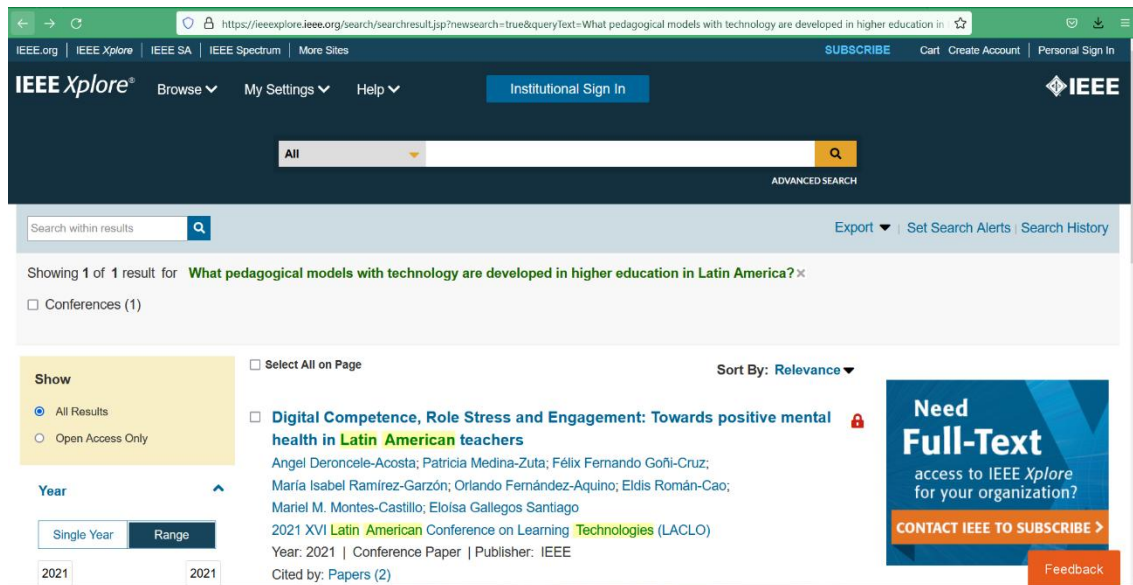
En WOS no se encontró ningún recurso con la pregunta de investigación planteada en inglés y en español; en la Figura 21 se puede apreciar lo antes mencionado.

Figura 21
Consulta en WOS



En la base de Datos IEEE se encontró 1 artículo referente a la pregunta propuesta en inglés (Ver Figura 22).

Figura 22
Consulta en la base de datos IEEE



En la base de datos ACM se encontraron 70 recursos con la pregunta de investigación en inglés y 35.147 en español (Ver Figuras 23 y 24).

Figura 23
Consulta ACM Pregunta de Investigación en inglés

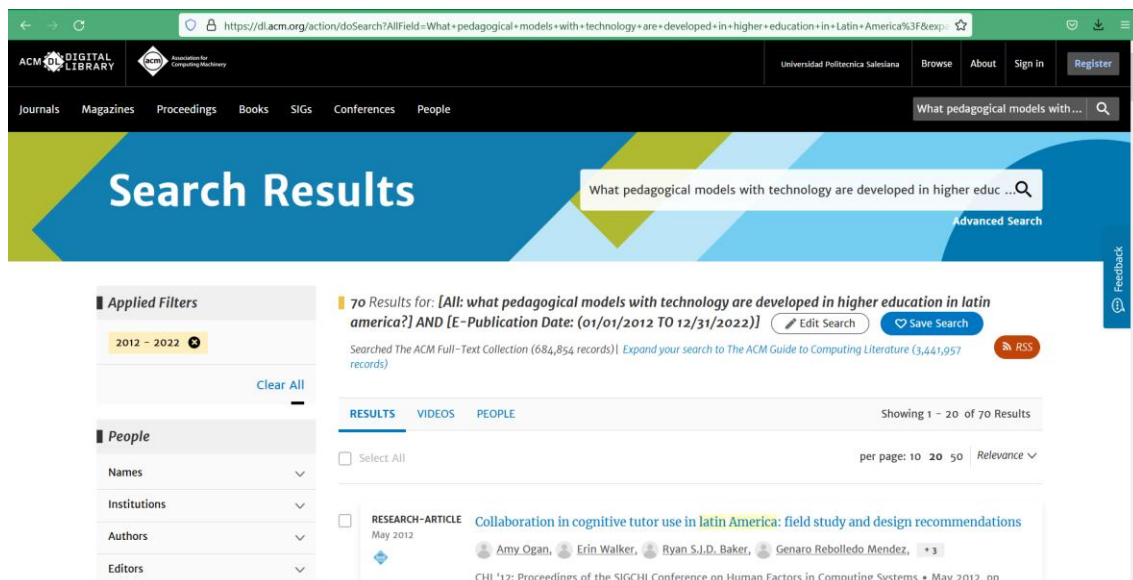
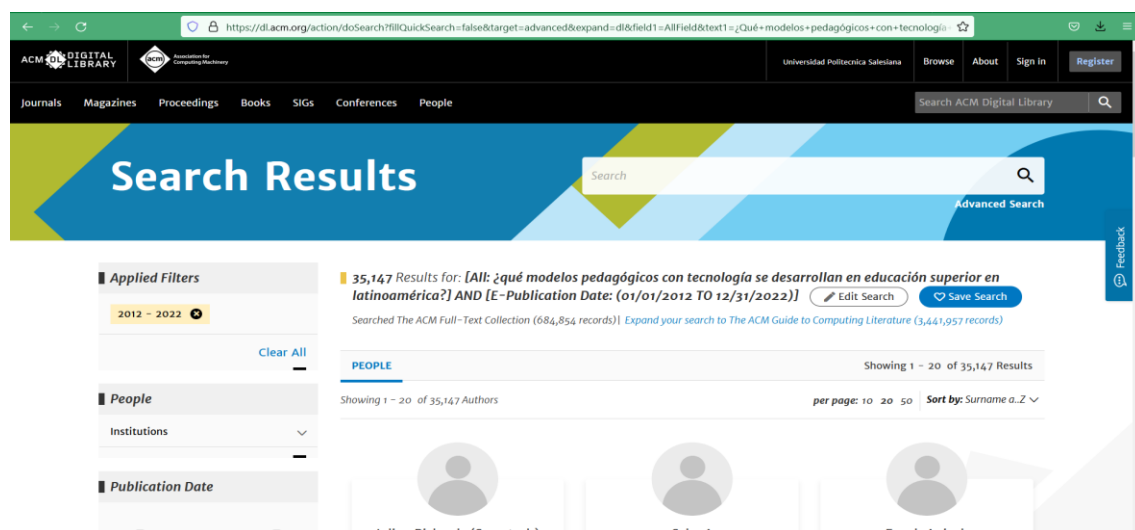
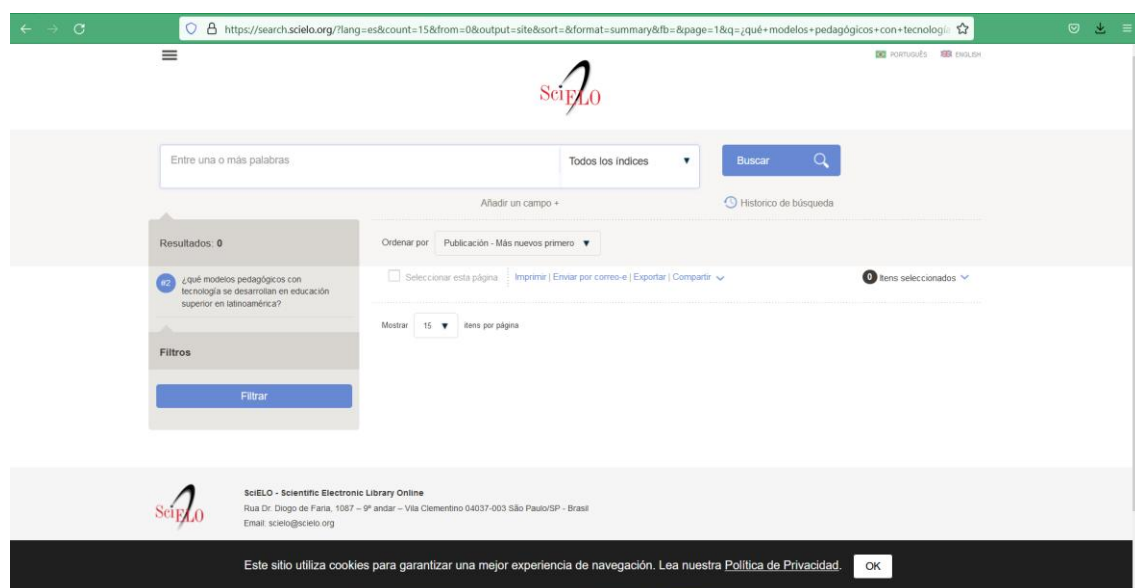


Figura 24
Consulta ACM Pregunta de Investigación en español



En la base datos Scielo no se encontró ningún recurso con la pregunta de investigación propuesta (Ver Figura 25).

Figura 25
Consulta base de datos Scielo



Entre las principales contribuciones que destacan de la revisión es importante reportar lo siguiente:

De acuerdo con Lai et al. (2022) los valores que sustentan la evaluación del uso de la tecnología en la educación se obtienen a través de las percepciones de especialistas de campo. El estudio plantea reflexiones críticas sobre el rigor del desarrollo de instrumentos de

evaluación en el campo de la tecnología educativa. La evaluación integral del uso de la tecnología es esencial para tomar decisiones sobre las estrategias de aprendizaje de la tecnología. Hay una falta de trabajo crítico que cuestione cómo se evalúa el uso de la tecnología en la educación, es necesario desarrollar instrumentos de evaluación en el campo de la tecnología educativa que se vean reflejados en los modelos pedagógicos que se basen en tecnologías.

Los estudios de educación superior, fuera de la formación docente, muestran que la tutoría y la consulta de tecnología individual fomentan la adopción de tecnología en los docentes; añaden que los desarrolladores académicos pueden usar tres enfoques para facilitar la creación de TPACK durante la consulta individual con la facultad: modelado tecnológico, realineación pedagógica y práctica de profundización (Koh,2020).

En los últimos años, las plataformas de seminarios web se han utilizado ampliamente en el aprendizaje en línea para proporcionar un ambiente de aprendizaje sincrónico en el que los estudiantes pueden interactuar y colaborar entre sí en tiempo real. Este enfoque se basa en el reconocimiento de que el aprendizaje es un proceso social, y que la interacción y la colaboración entre los estudiantes son fundamentales para un aprendizaje efectivo.

La adecuada integración de las TIC en los procesos educativos y formativos es actualmente uno de los mayores retos de la educación. El aula del presente es muy diferente a la de hace apenas unas décadas, las nuevas herramientas tecnológicas están transformando por completo sus características y actividades, y de ello surge la necesidad de plantear modelos pedagógicos que sean soportados por tecnologías.

Existen varios modelos pedagógicos que utilizan tecnología en educación superior, algunos de los más comunes son:

1. Modelo de aprendizaje en línea: Este modelo emplea internet y las redes sociales para brindar educación a distancia. Los estudiantes pueden acceder a los materiales de estudio y a las clases en línea en cualquier momento y lugar.

El aprendizaje en línea es automático, el modelo recibe datos de entrada de manera continua y actualiza sus parámetros en consecuencia.

2. Modelo de aprendizaje móvil: Utiliza dispositivos móviles como smartphones y tablets para brindar educación a distancia. Los estudiantes pueden acceder a los materiales de estudio y a las clases en cualquier momento y lugar, utilizando sus dispositivos móviles. Estos modelos son especialmente útiles para aplicaciones que requieren procesamiento de datos en tiempo real y para las que no es práctico enviar los datos a un servidor central de procesamiento. Los modelos de aprendizaje móvil pueden ser entrenados en la nube y luego se descargan en el dispositivo móvil para su uso, o pueden ser entrenados directamente en el dispositivo móvil.
3. Modelo de aprendizaje basado en proyectos: Este modelo utiliza proyectos prácticos y colaborativos para involucrar a los estudiantes en el proceso de aprendizaje. Los estudiantes trabajan en equipo para resolver problemas y completar proyectos, lo que les permite aplicar lo aprendido en situaciones reales. El modelo de aprendizaje basado en proyectos tiene un enfoque educativo en el cual los estudiantes aprenden a través de la realización de proyectos concretos. Este modelo se centra en el aprendizaje práctico y en la resolución de problemas, y permite a los estudiantes aplicar lo que han aprendido en situaciones similares a las que se enfrentarán en su vida laboral. El aprendizaje basado en proyectos también fomenta el pensamiento crítico, la colaboración y la creatividad.
4. Modelo de aprendizaje personalizado: Este modelo utiliza tecnologías de inteligencia artificial y aprendizaje automático para personalizar el aprendizaje para cada estudiante; se utiliza el historial de aprendizaje del estudiante para adaptar el contenido y el ritmo de aprendizaje a las necesidades individuales. Un modelo de aprendizaje personalizado es un algoritmo que se entrena para adaptarse a las preferencias individuales de un usuario.
5. Modelo de aprendizaje híbrido: Este modelo combina el aprendizaje en línea y el aprendizaje presencial. Los estudiantes asisten a clases presenciales y también

tienen acceso a materiales de estudio y actividades en línea. Un modelo de aprendizaje personalizado es un algoritmo que se entrena para adaptarse a las necesidades y preferencias individuales de un usuario.

3.7 Contexto de la educación superior en la enseñanza de la programación

La educación superior juega un papel importante en la enseñanza de la programación. A través de programas universitarios y posgrados, los estudiantes pueden adquirir habilidades y conocimientos especializados en programación, así como en otras áreas relacionadas como la informática, la ingeniería de software y la ciencia de datos. Estos programas suelen incluir cursos teóricos y prácticos, así como proyectos y oportunidades de prácticas profesionales para ayudar a los estudiantes a desarrollar habilidades prácticas y aplicar lo que han aprendido en situaciones reales. Además, la educación superior también puede proporcionar una red de contactos y recursos valiosos para ayudar a los estudiantes a encontrar oportunidades de trabajo y carreras en el campo de la programación.

La tendencia actual de las Instituciones de Educación Superior (IES), se enfoca en la competitividad académica en función del mejoramiento y aseguramiento de la calidad, la misma que se sujeta a una política de Estado. Cuando se habla de calidad en la educación superior es necesario mencionar la Conferencia Regional de la Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura (UNESCO, 2008), en la que se planteó que la calidad se define como la transformación del ser y del quehacer de la educación superior a su deber ser. En esta conferencia se definió que la calidad de la educación superior aparece vinculada a una problemática actual de extrema complejidad, que incluye: las falencias estructurales de los sistemas educativos, dadas por la escasa interrelación y coherencia entre sus distintos niveles; la alarmante disminución de los presupuestos destinados a satisfacer las crecientes necesidades; la necesidad de asumir procesos de innovación curricular acorde con las necesidades sociales; el continuo mejoramiento del proceso de enseñanza-

aprendizaje desde todos sus componentes, y el vertiginoso avance del conocimiento y del acceso a la información.

Arraya-Castillo (2013) argumenta: “la calidad del servicio es vista por las universidades como un componente de la estrategia para satisfacer las necesidades de sus clientes, ‘los estudiantes’, aunque estos no son los únicos clientes, pues también estarían el personal administrativo, los profesores, el gobierno, las familias, las empresas y la sociedad en general”(p.4).

En cuanto a los estudiantes como fuente de ventaja competitiva, son los que más han llamado la atención de las instituciones de educación superior que, además de preocuparse por desarrollar habilidades y destrezas requeridas por la sociedad, deben conocer cómo se sienten sus clientes acerca de su experiencia educacional (Arraya-Castillo,2013); en consecuencia, se están dando cuenta cada vez más de la importancia de la educación superior como una industria de servicios y están poniendo más énfasis en la satisfacción de las expectativas y necesidades de los estudiantes.

Es importante la presencia de modelos de calidad en la educación superior. El modelo de gestión de calidad educativa de Arcaro (1995) se compone de cinco pilares básicos: a) orientación al cliente, b) involucramiento total de toda la institución, c) establecimiento de medidas de desempeño, d) compromiso de la administración y e) implementación del programa de mejora continua en todos los indicadores señalados como prioritarios.

La gestión de la calidad en las universidades implica diseñar, estandarizar e implementar todas las tareas y procedimientos que garanticen el desarrollo efectivo de las funciones básicas de la institución, teniendo como referente su misión y las necesidades de la región en la que actúa, desde una visión prospectiva enmarcada en las tendencias nacionales e internacionales. Así mismo, comprende el diseño y aplicación de mecanismos de seguimiento, evaluación y mejoramiento a los procesos, y procedimientos establecidos, que brinden al sistema de gestión de calidad el dinamismo y adaptabilidad requeridos para

asegurar que su funcionamiento logre el nivel de excelencia demandado por la sociedad, sin caer en la rigidez de la homogenización. Además, requiere la afluencia de diferentes elementos, a saber (Tabla 5):

Tabla 5

Características de la gestión de la calidad educativa

-
- a. Políticas y lineamientos que orienten la actuación de la institución en la docencia, investigación, extensión y soporte administrativo.
 - b. Compromiso de todos (directivos y funcionarios en general) logrado mediante políticas de desarrollo del talento humano tendientes a consolidar planes, programas y acciones, con el fin de fortalecer la relación institución-funcionario desde el momento en que ingresa, durante su permanencia en la universidad y hasta cuando se retira. De tal forma que se sienta partícipe y beneficiario del sistema de gestión de calidad y vele por su funcionamiento y mantenimiento.
 - c. Infraestructura física y tecnológica para apoyar los procesos y procedimientos.
 - d. Definición, divulgación, estandarización y aplicación de procesos estratégicos (planeación, seguimiento, control y evaluación), misionales (formación, investigación, extensión), de apoyo (gestión de talento humano, recursos físicos, financiera, de sistemas, entre otros) y otros que se requieran, teniendo como base y horizonte las tendencias y necesidades del entorno en un marco prospectivo de desarrollo de la región y el país.
 - e. Diseño de mecanismos de control, medición, evaluación y mejora, pertinentes y necesarios para monitorear permanentemente que los procesos se cumplan de acuerdo con la misión y objetivos institucionales trazados en el proyecto institucional, los planes estratégicos y de gestión (León, 2008).

Nota. Gestión de la calidad educativa adaptada de (León,2008)

De acuerdo con De Pablos et al. (2019) :

La universidad constituye un sistema complejo que resulta clave para el desarrollo del conocimiento científico y la formación de profesionales. La integración digital que se está implantando a nivel global también se está produciendo en las instituciones universitarias. El uso de diversas herramientas tecnológicas en educación superior integra competencias y habilidades que unen a docentes, investigadores y estudiantes, los cuales generan conocimiento. En la actualidad se requiere modelos educativos abiertos y flexibles, al servicio del estudiante, que potencien el desarrollo de competencias aplicables a nivel profesional y vital; que permitan el reciclaje y acceso a la formación continua. Las herramientas que facilitan y obligan a pasar de un modelo a otro son las TIC (p.63).

Los principales interesados en la educación (estudiantes, profesionales de la educación, instituciones educativas y personal de TI) pueden beneficiarse de la CN, pero no todas aprovechan por igual sus ventajas. Algunas de estas posibilidades se refieren al aprendizaje y proceso de enseñanza, mientras que otras tratan diferentes aspectos de la educación, como la economía de TI o la gestión y operación de la tecnología.

La innovación de la calidad de la enseñanza y el aprendizaje en las universidades se ha convertido en algo primordial y, aunque es difícil realizar cambios en una institución, se debe tomar en cuenta que si no hay cambios no se puede avanzar en el conocimiento.

En el método tradicional de educación, el único medio para aprender es enseñar en el aula. Gradualmente, la educación derivó algunas soluciones como la distribución de material en papel y con el desarrollo de multimedia el material fue distribuido en discos compactos, lo que contribuyó a que los estudiantes tengan un mejor acceso a la información y se dio inicio a la educación no presencial (Tuncay, 2010). Con la ayuda de internet, y las grabaciones de audio y video, la enseñanza presencial se transmite a los alumnos y se proporciona acceso remoto a las clases; esto es la base para el aprendizaje mediante la Web (Cambria, 2016).

El enfoque de la educación, basado en la tecnología, tiene el potencial de mejorar el proceso de enseñanza, aprendizaje y evaluación (Al-Samarraie y Saeed, 2018). El entorno de aprendizaje habilitado por la tecnología proporciona una colección digitalizada de

artefactos que incluyen demostraciones y recursos con animaciones y visualizaciones (González-Martínez et al., 2015).

De acuerdo con Hillon-Salgado (2022) la programación en los espacios escolares ha sido fundamental porque los principales objetivos son la formación y desarrollo de habilidades de quienes la utilizan. De esta forma, contribuye a la actividad del estudiante desde el ámbito profesional y en la vida práctica. Hay investigadores que enfatizan que la programación de computadoras requiere el uso de técnicas de resolución de problemas. La falta de dominio de estas técnicas dificulta el aprendizaje de la programación (Tavares et al., 2012); así mismo, estos autores manifiestan que es importante que los estudiantes puedan resolver problemas de manera eficiente cuando trabajan en cooperación con compañeros, maestros y mentores.

La habilidad matemática se cita como necesaria y estrechamente relacionada con la capacidad de resolución de problemas. Cuando hablan con los maestros de programación, la mayoría de ellos afirman que los estudiantes no saben programar porque no saben cómo resolver problemas y no tienen suficientes antecedentes matemáticos. Otra habilidad importante dentro del contexto del aprendizaje de la programación es la “abstracción” que de acuerdo con Wing (2011) es el proceso de pensamiento más importante y de alto nivel en el pensamiento computacional utilizado para definir patrones, generalizar a partir de instancias específicas y parametrización para capturar propiedades esenciales comunes a un conjunto de objetos. De hecho, un programador necesita aplicar la abstracción al analizar problemas computacionales, así como crear instancias de conceptos y técnicas de programación abstracta para resolver problemas computacionales particulares.

La naturaleza abstracta de la programación, con argumentos similares: “Los estudiantes en la programación de aprendizaje necesitan imaginar y comprender muchos términos abstractos que no tienen equivalentes en la vida real: ¿cómo funciona una variable, un tipo de datos o una dirección de memoria relacionada con un objeto de la vida real? Estos conceptos son difíciles de entender”. En consecuencia, muchos estudiantes luchan por comprender, incluso, los conceptos de programación más básicos (Giraffa et al., 2013,p.18).

La sobrecarga de aprender la sintaxis y la semántica de un lenguaje de programación al mismo tiempo, las dificultades para combinar el conocimiento nuevo con el anterior y desarrollar sus habilidades generales de resolución de problemas son una preocupación permanente. De acuerdo con Giraffa et al. (2013) a los estudiantes que tienen habilidades adecuadas para resolver problemas, y logran formular una solución a un problema de programación en términos de un pseudocódigo, les resulta difícil convertir el pseudocódigo en un programa informático sintácticamente correcto. En este contexto, es importante considerar el error de sintaxis como parte del proceso de aprendizaje y ayudar a los estudiantes a aprender a superar las equivocaciones (p.20).

Los errores de sintaxis son un área relevante de la investigación pedagógica de programación. Los programadores experimentados rara vez cometen errores de sintaxis, y cuando lo hacen, tienen estrategias claras para corregirlos muy rápidamente. Sin embargo, los errores de sintaxis son importantes para los programadores novatos, corregirlos es un proceso que lleva mucho tiempo, y a menudo conduce a un comportamiento de depuración aleatorio, también influido por el hecho de que los estudiantes no entienden el mensaje del compilador.

Otro aspecto fundamental en el proceso de aprendizaje de programación es el razonamiento algorítmico (un conjunto de habilidades que están conectadas a la construcción y comprensión de algoritmos) para analizar problemas, especificar un problema con precisión, encontrar acciones básicas adecuadas al problema y construir un algoritmo correcto para un problema usando las acciones básicas.

La depuración se describe como una actividad compleja, que requiere dominar varios aspectos para ello los estudiantes deben comprender el dominio del problema, conocer conceptos de programación rudimentarios y comprender al menos un lenguaje de programación lo suficientemente bien como para leer y escribir instrucciones, comprender la lógica del programa previsto y ser capaces de rastrear y corregir errores. (Giraffa et al., 2013).

Algunos estudiantes pueden tener motivación, pero la confianza en sí mismos puede verse bloqueada, ya que este tipo de estudiantes necesita que el maestro trabaje más cerca de ellos para desarrollar su autoestima en el proceso de aprendizaje. Parece que la motivación preocupa mucho a los maestros, por el alto impacto en los problemas de aprendizaje. Se necesitan enfoques pedagógicos para ayudar a los maestros a mantener motivados a los estudiantes, por ello es importante que el maestro utilice las herramientas para un adecuado proceso de enseñanza y aprendizaje.

Para la enseñanza de la programación a nivel universitario se utilizan diferentes metodologías y actividades, así como diferentes tecnologías computacionales, entre ellas la CN. La CN proporciona a los estudiantes y maestros un gran número y variedad de aplicaciones en línea que pueden ser empleadas para soportar una amplia gama de escenarios de aprendizaje. Así mismo, la computación en la nube puede ayudar a superar muchos problemas en las escuelas al tener los tipos correctos de tecnologías, incluidos hardware y software (Brush et al., 2008,p.113).

Cabero (2014) indica las características más significativas del proceso de enseñanza-aprendizaje en el ámbito de la educación superior, se presentan en la tabla 6.

Tabla 6
Características proceso de enseñanza–aprendizaje

Abierto.
Social y colaborativo.
Mezcla de lo real y lo virtual.
Visual.
Descontextualizado y ubicuo.
Personalizado.
Móvil

Nota. Características más significativas del proceso de enseñanza-aprendizaje

Dentro de este contexto, si queremos realizar acciones de CN de calidad no es suficiente con centrarse en los aspectos meramente tecnológicos e instrumentales. Los docentes deben poner énfasis en las variables de tipo educativo y organizativo. Estas van desde la forma en la que se estructuran los contenidos, las metodologías que se aplicarán, la planificación de las actividades que realizarán los estudiantes y los roles que como docentes y tutores virtuales desempeñaremos.

3.8 Plataformas digitales en la educación superior

Las plataformas digitales en la educación superior son herramientas tecnológicas utilizadas para brindar acceso a contenido educativo y permitir la comunicación y colaboración entre estudiantes y profesores. Estas plataformas pueden incluir sistemas de gestión del aprendizaje (LMS), videoconferencia, herramientas de colaboración en tiempo real y aplicaciones de evaluación. Algunos ejemplos populares de plataformas digitales en la educación superior incluyen Blackboard, Canvas y Coursera.

El desarrollo constante de las TIC ha provocado que la sociedad gire hacia un contexto digital basado en conexiones. La educación superior ha evolucionado, cambiando tanto la manera de aprender como la de enseñar. La tecnología es el eje transversal en todo proceso educativo en línea, por lo que la presente investigación analiza las plataformas digitales como un nuevo estilo de aprendizaje en la educación superior (Lagos, 2018,p139). El aprendizaje mediante una plataforma educativa virtual ofrece ventajas en cuanto al entorno de la flexibilidad de horarios, mayor diversidad de recursos y la posibilidad de colaborar e intercambiar experiencias en el desarrollo de competencias digitales.

La educación en la actualidad es la encargada no solo de transmitir conocimientos e información, sino de proporcionar los medios, aptitudes y habilidades para producirlos, y utilizarlos en la práctica pedagógica, creando nuevos escenarios donde desarrollan el proceso educativo. Desde los comienzos de las acciones formativas, con el uso de

plataformas digitales, hasta la actualidad se han efectuado diversos trabajos (investigaciones, reflexiones teóricas, metaanálisis y estudios de buenas prácticas) que permiten apuntar, tanto algunos aspectos a contemplar para la realización de acciones formativas de calidad, como para detectar los errores y dificultades más significativas que se pueden encontrar para la realización de acciones formativas.

A continuación, se presentan algunas sugerencias que se pueden aplicar al usar e interactuar en plataformas digitales:

La acción del docente como tutor virtual es crucial para el aprendizaje de los estudiantes en entornos virtuales. El docente juega un papel importante en la orientación, motivación y retroalimentación de los estudiantes, lo que contribuye a mejorar su experiencia de aprendizaje en línea.

La calidad del funcionamiento y la facilidad de uso de los dispositivos tecnológicos son factores importantes para garantizar la efectividad de los entornos de formación virtual. Los dispositivos deben ser confiables y fáciles de usar para que los estudiantes puedan centrarse en el contenido de la formación y no en resolver problemas técnicos con el dispositivo. Además, es importante asegurar que la tecnología utilizada sea compatible con las necesidades de accesibilidad de los estudiantes.

La incorporación al diseño instruccional, que realice el docente, de diferentes herramientas de comunicación sincrónicas y asincrónicas y de la Web 2.0, mejora y facilita el aprendizaje en los estudiantes. Lo verdaderamente significativo del e-learning es crear un nuevo entorno de comunicación. La motivación que los estudiantes tengan para participar en las acciones e-learning, y la capacidad para la autorregulación de su aprendizaje condiciona el rendimiento alcanzado. Se deben establecer mecanismos para que los estudiantes reflexionen sobre su propio aprendizaje.

La computación en la nube pública en adelante (CNP) proporciona infraestructura, servicios y software a través de la red y ofrece atractivas ventajas a las instituciones de

educación superior. En los últimos años, la CN se ha debatido cada vez más. Esta es una tendencia relativamente nueva del desarrollo de la industria de TI, centrada en los usuarios e impulsada por el uso cada vez mayor de dispositivos móviles, como computadores portátiles, tabletas y teléfonos inteligentes (Macias, 2019,p.35). Así mismo, Quezada et al. (2022) explica que el uso de herramientas de CN permite generar un ambiente de mayor interactividad entre docentes y estudiantes en formación, y un proceso educativo adecuado.

El aumento de la oferta de educación superior es una necesidad básica de los países desarrollados y emergentes, requiere inversiones crecientes y continuas. La oferta de educación superior, a través del aprendizaje basado en internet, es una de las formas más eficientes para la masificación de esta oferta, ya que permite una amplia cobertura y menores costes. En este escenario, destacamos Moodle como un entorno abierto y de bajo costo para el aprendizaje a distancia.

Su utilización puede amplificarse mediante la adopción de las TIC y CN, que permiten la virtualización de los sitios de Moodle, reduciendo costes, facilitando la gestión y aumentando su capacidad de servicio.

Finalmente, es cierto que ampliar la oferta de educación superior es importante para el desarrollo económico y social de los países; sin embargo, esto requiere inversiones significativas en infraestructura y recursos humanos para asegurar que la educación proporcionada sea de alta calidad. Además, es importante asegurar que la educación superior sea accesible y que las personas tengan las habilidades y conocimientos necesarios para aprovechar las oportunidades de empleo y mejorar su calidad de vida.

3.9. Competencias digitales

Las competencias digitales son habilidades, conocimientos y destrezas relacionadas con el uso de tecnologías digitales. Esto incluye habilidades para el manejo de dispositivos y

aplicaciones digitales, habilidades para buscar, evaluar y utilizar información en línea, y para comunicarse y colaborar en línea. Estas habilidades son cada vez más importantes en un mundo digitalizado y son esenciales para el éxito en muchos campos laborales y personales.

De acuerdo con Bautista et al. (2006) la competencia digital implica la confianza y el uso crítico de la Tecnología de la Sociedad de la información (TSI) para el ocio y la Comunicación en el trabajo. Se basa en habilidades básicas en TIC: el uso de computadoras para acceder, evaluar, almacenar, producir, presentar e intercambiar información, y para comunicarse y participar en redes de colaboración, a través de Internet (p.36).

Los estudios y experiencias que se han realizado en los últimos tiempos en la formación de competencias digitales sugieren distintos aspectos que debemos contemplar a la hora de su puesta en funcionamiento con acciones de calidad, al respecto podemos destacar los siguientes:

- Los docentes y los estudiantes deben tener unas mínimas competencias tecnológicas para desenvolverse en los entornos virtuales de formación. Esto incluye habilidades básicas en el uso de dispositivos y aplicaciones digitales, así como conocimientos en seguridad y privacidad en línea. También es importante que los docentes estén familiarizados con las herramientas pedagógicas y metodologías específicas para la enseñanza en línea, por ello deben estar capacitados en el uso de estas herramientas y metodologías para adaptarse mejor al cambio y brindar una educación efectiva a los estudiantes.
- La acción desarrollada por el docente como tutor virtual es clave para que los alumnos aprendan en los entornos virtuales. El docente debe ser capaz de proporcionar retroalimentación y orientación, fomentar la colaboración y el trabajo en equipo, y asegurar que los estudiantes tengan acceso a los recursos y herramientas necesarios para el aprendizaje en línea. Además, es importante que el docente establezca una comunicación efectiva y mantenga un ambiente de aprendizaje seguro y positivo.

- La calidad del funcionamiento y la facilidad de uso de los dispositivos tecnológicos utilizados en la formación virtual son factores clave para garantizar una experiencia de aprendizaje eficaz y satisfactoria. Esto incluye tanto la elección de las herramientas adecuadas como la capacitación del personal en su uso.
- Respecto al diseño de los materiales que se incorporen a este tipo de formación, se muestran más eficaces aquellos que reúnen diferentes recursos audiovisuales, auditivos y multimedia. Estos recursos ayudan a involucrar a los estudiantes de diferentes maneras y les permite aprender de forma más activa y participativa, además pueden ayudar a mejorar la retención de la información y aumentar la motivación del estudiante.
- La creación de un ambiente de aprendizaje colaborativo y participativo es esencial en un entorno de formación virtual. El docente debe ser capaz de fomentar la interacción entre los estudiantes y proporcionar un ambiente seguro y receptivo para el diálogo y el intercambio de ideas. Esto ayudará a desarrollar habilidades sociales y de comunicación, además, aumentará la motivación y el compromiso de los estudiantes con el aprendizaje.
- La percepción del estudiante de que no hay nadie en la red (en la acción formativa) condiciona el resultado que se obtenga. Estimular el contacto, estudiante-profesor, aumenta la satisfacción en la experiencia de formación virtual.
- La creencia que tenga el docente sobre la calidad de la formación virtual condiciona el resultado que obtengan los alumnos en el e-learning. Los docentes que creen en las ventajas de la educación virtual pueden estar más motivados para preparar y presentar contenido de manera efectiva en línea, mientras que aquellos que tienen dudas o prejuicios sobre la educación en línea pueden no esforzarse tanto o presentar contenido de manera menos efectiva. Por lo tanto, es importante que los docentes estén capacitados y motivados para enseñar en línea y así garantizar que los estudiantes tengan una experiencia educativa de alta calidad.
- La incorporación al diseño instruccional que realice el docente de diferentes herramientas de comunicación sincrónicas, asincrónicas, y de la Web 2.0 mejora y

facilita el aprendizaje en los estudiantes. Lo verdaderamente significativo del e-learning es crear un nuevo entorno de comunicación.

- Las prácticas de formación en el e-learning son las que estimulan la cooperación entre estudiantes que se desenvuelven dentro de una estrategia de aprendizaje colaborativo. Las prácticas de formación en el e-learning incluyen el uso de plataformas en línea para la creación y distribución de contenido educativo, así como la comunicación y colaboración entre estudiantes y profesores a través de internet. Esto puede incluir el uso de videos, presentaciones, juegos educativos, discusiones y evaluaciones en línea. También se pueden utilizar herramientas de seguimiento y análisis para evaluar el rendimiento de los estudiantes y mejorar el contenido.
- Estimular un aprendizaje activo, mediante la aplicación de estrategias y metodologías específicas, es clave para alcanzar un aprendizaje de calidad. La realización de tareas y e-actividades se convierte en un elemento clave para que los estudiantes adquieran conocimientos y alcancen altos niveles de aprendizaje en los entornos de formación e-learning.
- El ofrecer una retroalimentación rápida a las demandas de los estudiantes es percibido por ellos como un entorno de formación de calidad, lo que aumenta su nivel de satisfacción por participar en la experiencia e incrementa su rendimiento.
- Respetar los diferentes estilos y capacidades de los alumnos. Esto incluye adaptar la enseñanza a diferentes habilidades de aprendizaje, implementar métodos de enseñanza variados y crear un ambiente de aprendizaje inclusivo y respetuoso. Además, reconociendo las diferencias individuales de los estudiantes, y adaptando la enseñanza en consecuencia, se puede mejorar su motivación y rendimiento.
- Los alumnos prefieren una actitud más reactiva de los tutores virtuales. Esto significa que los tutores deben estar disponibles para responder preguntas y proporcionar ayuda en el momento en que se necesite, en lugar de seguir un plan predeterminado o esperar que los estudiantes busquen apoyo. Esto puede mejorar la experiencia de aprendizaje de los estudiantes que contarán con ayuda de manera oportuna.

- La estructuración de los contenidos de aprendizaje tiene un gran impacto en cómo los estudiantes interactúan y aprenden. Una estructuración clara y lógica puede ayudarlos a comprender mejor el material y a retenerlo de manera más efectiva. Por otro lado, una estructuración confusa o desordenada puede dificultar el aprendizaje y causar confusión.
- La motivación que los estudiantes tengan para participar en las acciones de e-learning, y su capacidad de autorregulación del aprendizaje, condiciona el rendimiento alcanzado (Bautista et al., 2006,p.47).

Si queremos realizar acciones en el entorno de las competencias digitales para que estas sean de calidad, no es suficiente con centrarse en los aspectos meramente tecnológicos e instrumentales. Los docentes deben poner énfasis en las variables de tipo educativo y organizativo. Estas van desde la forma en la que se estructuran los contenidos, las metodologías que se aplicarán, la planificación de las actividades que ejecutarán los estudiantes, y los roles que como docentes y tutores virtuales se desempeñen.

Finalmente, es importante mencionar que la competencia digital presupone el interés por las tecnologías digitales y su utilización con familiaridad y espíritu crítico, responsable de aprender, trabajar y participar en la sociedad. Incluye la alfabetización informática y digital, la comunicación y la colaboración, la alfabetización mediática, la creación de contenidos digitales (incluida la programación), la seguridad, las cuestiones de propiedad intelectual, la solución de problemas y el pensamiento crítico.

3.10 Dimensiones de la pedagogía digital

La pedagogía digital se refiere al uso de tecnologías digitales en el proceso de enseñanza-aprendizaje. Algunas dimensiones importantes de la pedagogía digital incluyen:

1. Diseño instruccional. El diseño instruccional es el proceso de planificar, desarrollar y evaluar la instrucción y el aprendizaje. Incluye la identificación de los objetivos de aprendizaje, la selección y organización del contenido, la selección de los medios y métodos para presentar el contenido, y la evaluación del aprendizaje. El diseño instruccional se utiliza en una variedad de entornos de aprendizaje, como la educación formal, la capacitación en el lugar de trabajo y la educación en línea.
2. Comunicación y colaboración: cómo se utilizan las tecnologías digitales para facilitar la comunicación y el trabajo en equipo entre estudiantes y profesores.
3. Evaluación y retroalimentación: cómo se utilizan las tecnologías digitales para evaluar el progreso del estudiante y proporcionar retroalimentación en tiempo real.
4. Accesibilidad e inclusión: cómo se utilizan las tecnologías digitales para garantizar que el contenido educativo y las actividades de aprendizaje sean accesibles para una variedad de estudiantes con necesidades especiales.
5. Integración en el currículo: cómo se utilizan las tecnologías digitales para integrar el aprendizaje en línea en el currículo general y mejorar el aprendizaje de los estudiantes en áreas como las ciencias, las matemáticas y las artes.

En este mismo contexto, la “pedagogía digital” se refiere al uso de elementos electrónicos para mejorar o cambiar la experiencia de la educación (Croxal, 2012).

Ilaltdinova et al., (2019) presentó una definición más detallada, refiriéndose a la pedagogía digital:

Como la habilidad de incorporar tecnologías digitales en la enseñanza para que mejoren el aprendizaje, la formación, la evaluación y el plan de estudios. Por lo tanto, la pedagogía digital también puede considerarse un uso pedagógico de las tecnologías digitales. Este modelo de pedagogía digital comprende tres dimensiones: 1) orientación pedagógica, que depende de las metas curriculares, del rol del docente en relación con las prácticas de enseñanza, y del rol de los estudiantes en relación con las prácticas de aprendizaje, 2) prácticas pedagógicas, se refiere a los métodos que utiliza el maestro para enseñar y 3) competencias pedagógicas

digitales enfocadas a las habilidades que los docentes necesitan para integrar con éxito las tecnologías digitales en la enseñanza. Las competencias pedagógicas de los educadores se dividen en cuatro segmentos: recursos digitales, enseñanza y aprendizaje, evaluación y empoderamiento de los alumnos (p.10).

Apelgren y Giertz (2010) argumentan que la competencia pedagógica comprende seis aspectos: actitud, conocimiento, habilidad, adaptación a la situación, perseverancia y continuo desarrollo. Así mismo, From (2017) adoptó un enfoque similar al definir “competencia digital pedagógica” (PDC) como una nueva dimensión de la pedagogía de los maestros contemporáneos, con tres niveles: interacción, curso y organización.

Mishra y Koehler (2006) desarrollaron un marco de conocimiento para la integración de tecnología. El conocimiento del contenido pedagógico tecnológico (TPACK) intenta describir cómo los tres componentes principales del conocimiento de los maestros se relacionan. El primer componente, el conocimiento del contenido, describe lo que saben los profesores sobre el tema que enseñan. El segundo componente, el conocimiento pedagógico, se refiere a la comprensión de los profesores de los procesos, métodos de enseñanza y aprendizaje. El tercer componente, conocimiento tecnológico, se refiere a la habilidad de los maestros para aplicar la tecnología productivamente en el trabajo y en sus vidas cotidianas.

Expresado lo previo dentro del modelo pedagógico que se propone en la presente tesis doctoral, se consideraron cuatro dimensiones:

- 1. Orientación Pedagógica.** Una disciplina educativa que se dedica al estudio de los procesos de enseñanza-aprendizaje y al desarrollo de estrategias y técnicas para mejorar la calidad de la educación. Incluye la planificación, implementación y evaluación de programas educativos, así como el diseño de materiales didácticos y la formación de docentes. La orientación pedagógica también se ocupa de temas relacionados con la orientación de estudiantes y la atención a las necesidades

individuales de aprendizaje. En el caso de nuestro modelo lo enfocaremos desde tres aristas:

- La percepción del docente sobre cómo los estudiantes aprenden programación en el contexto de las ciencias de la computación, esto puede variar dependiendo del docente y del contexto específico en el que se encuentre. Sin embargo, algunos docentes pueden percibir que los estudiantes aprenden mejor a través de la práctica y la resolución de problemas, mientras que otros pueden creer que una enseñanza teórica es más efectiva. También pueden influir en la percepción del docente el nivel de experiencia de los estudiantes y el tipo de programación que se está enseñando.
- Metas curriculares basadas en el perfil profesional del Ingeniero en Informática. Las metas deben incluir habilidades y conocimientos en áreas como programación, bases de datos, redes de computadoras, inteligencia artificial, ciencia de datos, arquitectura de software, sistemas operativos, seguridad informática y proyecto de sistemas. Además, deben fomentar el desarrollo de habilidades blandas como la capacidad de trabajar en equipo, la comunicación efectiva y la capacidad de pensar críticamente.
- Rol del estudiante en el proceso de aprendizaje.

2. Prácticas Pedagógicas. Las prácticas pedagógicas son los métodos y estrategias utilizadas por los maestros y educadores para enseñar a los estudiantes. Estas pueden incluir técnicas de enseñanza activa, enfoques basados en proyectos, aprendizaje basado en la investigación y enseñanza personalizada. El objetivo es desarrollar habilidades y conocimientos en los estudiantes de manera efectiva y significativa.

En el contexto de nuestra investigación consideraremos:

- a. Métodos de enseñanza que el docente utiliza en las clases de programación, de ello se toman en cuenta los siguientes factores: Prácticas fundamentales

de las herramientas de CN y CA, que se enfocarán a las ideas iniciales de los docentes sobre la incorporación de las herramientas antes descritas.

- b. Desarrollo de prácticas de programación considerando la inclusión de herramientas de CN y CA en los currículos. Algunas de las prácticas recomendadas son:

Asegurar la accesibilidad de los materiales de estudio y de las herramientas de programación, lo que permitirá a los estudiantes de ingeniería informática acceder y utilizarlas.

Para asegurar la accesibilidad a los materiales de estudio y a las herramientas de programación, se pueden tomar varias medidas:

Utilizar formatos de archivo abiertos y estándares para los materiales de estudio, lo que garantiza que estén disponibles para todos los estudiantes independientemente del sistema operativo o dispositivo que utilicen.

Proporcionar acceso a las herramientas de programación a través de una plataforma en línea, lo que permite a los estudiantes acceder a ellas desde cualquier lugar con conexión a Internet.

Ofrecer alternativas para estudiantes con discapacidades, como subtítulos en videos o materiales de estudio en formatos de audio. Proporcionar soporte técnico para ayudar a los estudiantes a configurar y utilizar las herramientas de programación.

Asegurarse de que el material de estudio y las herramientas de programación estén disponibles en varios idiomas para atender a los estudiantes.

Incorporar en los currículos ejemplos y casos de estudio que representen a diferentes grupos.

Proporcionar capacitación y recursos para ayudar a los estudiantes, y profesores, a comprender mejor las necesidades y desafíos de las personas con discapacidad, así como de los grupos culturales diversos.

Fomentar una cultura de inclusión y diversidad en el aula, promoviendo un ambiente de aprendizaje seguro y respetuoso para todos los estudiantes.

Asegurar la inclusión de los estudiantes con características específicas de aprendizaje, y de los grupos culturales diversos, en el proceso de diseño y desarrollo de software.

Promover la formación continua de los profesores y estudiantes, para que puedan mantenerse actualizados en cuanto a las últimas tendencias y avances del área de la programación.

c. Prácticas de herramientas de CN basadas en las habilidades de programación, las que incluyen el uso de lenguajes de programación para automatizar tareas en la nube: configuración de infraestructura, despliegue de aplicaciones y monitoreo de recursos. Esto puede incluir el uso de herramientas como Terraform, para configurar infraestructura en proveedores de nube como Amazon Web Services (AWS), Microsoft Azure, el uso de lenguajes de programación como Python o JavaScript para crear scripts automatizados que realicen tareas específicas en la nube.

d. Prácticas pedagógicas en las que el docente comprende el uso de las herramientas de CN y la CA que permitan la participación de los estudiantes en actividades basada en el desarrollo de proyectos.

3. Competencias pedagógicas. Son las habilidades y conocimientos que un docente debe tener para llevar a cabo su labor educativa de manera efectiva. Estas pueden incluir habilidades para planificar y diseñar lecciones, utilizar diferentes estrategias de enseñanza, adaptarse a las necesidades de los estudiantes, evaluar el aprendizaje y

promover un ambiente de aprendizaje positivo. Las competencias pedagógicas son esenciales para el éxito de un docente y para guiar adecuadamente el aprendizaje de sus estudiantes.

En el modelo propuesto se enfoca en las habilidades que los docentes necesitan para integrar con éxito las herramientas de CN en el proceso de enseñanza-aprendizaje. Así mismo, se consideran las habilidades que debe tener el docente: actitud, conocimiento, destreza, adaptación, perseverancia y desarrollo continuo. En este enfoque, los estudiantes trabajan en equipo para investigar, planificar, diseñar, implementar y evaluar proyectos relacionados con temas relevantes para ellos. Los docentes actúan como facilitadores, brindando orientación y recursos para ayudar a los estudiantes a alcanzar sus metas. Estas prácticas impulsan a los estudiantes a desarrollar habilidades como el pensamiento crítico, la colaboración y la creatividad, y los prepara para el mundo laboral.

4. Conocimiento tecnológico pedagógico. Se refiere a la combinación de conocimiento sobre tecnologías educativas, la enseñanza y el aprendizaje. Es una disciplina interdisciplinaria que se enfoca en el uso de las tecnologías para mejorar el aprendizaje y el desarrollo de habilidades. Incluye el uso de herramientas tecnológicas, estrategias pedagógicas y metodologías para la implementación de tecnologías educativas en el aula y en el proceso de enseñanza-aprendizaje. En esta dimensión se consideran los siguientes elementos:

- a. Conocimiento pedagógico de la enseñanza de la programación que se refiere a la comprensión de cómo los estudiantes aprenden y cómo enseñar de manera efectiva la programación. Esto incluye el uso de metodologías de enseñanza adecuadas, la creación de un ambiente de aprendizaje positivo y la adaptación del contenido a diferentes estilos de aprendizaje. También se requiere una comprensión de los conceptos fundamentales de la programación y cómo enseñarlos de manera clara y concisa.

- b. El conocimiento técnico de programación se refiere a las habilidades y conocimientos necesarios para escribir, depurar y mantener un código de programación de manera eficiente. Esto incluye el conocimiento de lenguajes de programación específicos, estructuras de datos y algoritmos, así como las mejores prácticas para diseñar, probar y optimizar el código. Además de las habilidades técnicas, también se espera que un programador tenga una comprensión sólida de los principios fundamentales de la informática y sea capaz de resolver problemas de manera eficaz.
- c. La CA y CN son dos áreas relacionadas con la tecnología que pueden apoyar una variedad de contenidos curriculares. La computación en la nube permite el acceso remoto a aplicaciones y recursos, lo que facilita la colaboración y el trabajo en equipo en línea. Puede ser utilizada en áreas como la educación, la ingeniería, la investigación y la administración empresarial.

La CA se enfoca en el uso de la tecnología para medir y responder a los sentimientos y emociones de las personas. Puede ser utilizada en áreas como la psicología, la educación, la salud y la atención al cliente para mejorar la interacción humana con la tecnología. Algunos ejemplos de contenidos curriculares que podrían ser soportados por la computación en la nube y la CA incluyen:

1. Educación en línea y colaboración a distancia
2. Análisis de datos y ciencia de datos
3. Inteligencia artificial y aprendizaje automático,
4. Psicología y salud mental
5. Interacción humano-computadora.

Sin embargo, estos son solo algunos ejemplos; hay muchas otras áreas donde estas tecnologías pueden ser aplicadas. Así mismo se consideran los componentes de TPACK como uno de los elementos del modelo que se plantea en el capítulo 7. El modelo pedagógico

implica más que el uso de las herramientas de CN y la CA en el aula, el uso de las herramientas de CN debe basarse en la resolución de problemas y el desarrollo de habilidades de programación con un ambiente interactivo en el cual los alumnos sean creadores de su conocimiento al interactuar con el uso de las herramientas de CN y los principios de la CA.

El modelo pedagógico que se propone en la presente investigación busca qué integrar a la CN y CA, ya que ofrece varias ventajas en el ámbito educativo, como la accesibilidad y la escalabilidad. Los estudiantes y profesores pueden acceder a recursos y aplicaciones en línea desde cualquier lugar y en cualquier momento, lo que permite una mayor flexibilidad en el aprendizaje. Además, la CN Y CA permite a las instituciones educativas escalar fácilmente su capacidad para manejar un gran número de estudiantes y cursos. También puede ayudar a reducir los costos de hardware y software, ya que las aplicaciones y los datos se alojan y se administran en servidores remotos en lugar de dispositivos individuales.

Capítulo cuatro: La enseñanza superior de la computación en Ecuador

4.1. Contexto de las Ciencias de la Computación e Informática en el Ecuador

La informática hizo posible la automatización de tareas y procesos en muchas áreas de la economía, lo que lleva a una mayor eficiencia y productividad. Los sistemas informáticos realizan cálculos complejos en segundos, procesan grandes cantidades de datos y realizan tareas repetitivas de forma rápida y precisa. Esto permite a las empresas ahorrar tiempo, reducir costos y aumentar la capacidad de producción. Así mismo, la informática ha facilitado la comunicación dentro y alrededor del mundo.

Las empresas pueden comunicarse instantáneamente por correo electrónico, chat y videoconferencia para tomar decisiones comerciales y resolver problemas. Además, la conectividad global facilita el comercio electrónico y la expansión del mercado, brindando oportunidades para el crecimiento económico. La tecnología de la información fomenta la innovación y el desarrollo de nuevos productos y servicios. Con las computadoras, las empresas pueden realizar investigaciones, crear prototipos, simular escenarios y probar ideas de forma rápida y económica. Fomenta la innovación y el espíritu empresarial, además del crecimiento económico a través de la creación de empleo e ingresos.

“ El avance de la industria 4.0 se debe en gran medida al desarrollo de diferentes dispositivos, que juegan un papel protagónico, debido al éxito de los sistemas de cómputo en el procesamiento de información ” (Mitchell et al.,2009,p.999).

En Ecuador, la enseñanza de la informática y ciencias de la computación ha adquirido cada vez más importancia en los últimos años debido al reconocimiento de su relevancia en el desarrollo tecnológico y económico del país. Desde la educación básica hasta la educación superior, se imparten asignaturas relacionadas con la informática y la tecnología de la información. Esto garantiza que los estudiantes tengan la oportunidad de adquirir conocimientos y habilidades en el campo de la informática desde temprana edad.

Existen planes y programas de estudio específicos para la enseñanza de la informática en Ecuador. Estos programas están diseñados para reflejar los avances tecnológicos y las necesidades del mercado laboral. Se enfocan en áreas como programación, bases de datos, redes informáticas, seguridad informática e ingeniería de software, entre otras disciplinas afines. Existen varias instituciones educativas en el Ecuador, tanto públicas como privadas, que ofrecen programas de capacitación en computación. Desde escuelas técnicas hasta en universidades, los estudiantes pueden adquirir habilidades y conocimientos informáticos, en especial de programación. Algunas universidades también ofrecen títulos profesionales en informática, ingeniería de sistemas y tecnología de la información.

Además de la educación formal, en Ecuador también se han desarrollado programas de capacitación y certificaciones en informática. Estos programas brindan oportunidades para que los profesionales y estudiantes adquieran habilidades específicas en áreas como programación, análisis de datos, ciberseguridad, entre otras. Las certificaciones reconocidas internacionalmente, como las ofrecidas por Microsoft, Cisco u Oracle también son valoradas en el mercado laboral.

El gobierno ecuatoriano ha promovido el acceso a la tecnología y la informática en diferentes sectores de la sociedad. Se han implementado programas y proyectos para llevar la conectividad a áreas rurales, facilitar el acceso a dispositivos tecnológicos y promover el uso de herramientas informáticas en diferentes ámbitos, como la administración pública, la educación y los emprendimientos. Sin embargo, en el Ecuador se ha reconocido la importancia de la enseñanza de la informática y se han dado pasos para promover su enseñanza en los diferentes niveles educativos. Con esto se pretende preparar a estudiantes y profesionales para resolver problemas tecnológicos y aprovechar las oportunidades que brindan las tecnologías de la información en el desarrollo de la economía del país. En este contexto, se pone en consideración las 41 universidades ecuatorianas que ofertan las carreras relacionadas con las Ciencias de la Computación.

En la tabla 7 se presenta la lista de Instituciones de Educación Superior del Ecuador que ofertan la Carrera de Ciencias de la Computación, Informática y afines, en las cuales se imparte las asignaturas de programación.

Tabla 7

Universidades Ecuatorianas que ofertan carreras relacionadas a las Ciencias de la Computación

	Universidad	Título	Enlace
1	Escuela Politécnica Nacional	Ingeniero en Ciencias de la Computación	https://www.epn.edu.ec/
2	Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López	Ingeniero en computación	http://www.espam.edu.ec/
3	Universidad Técnica de Ambato	Ingeniero de Sistemas Computacionales e Informáticos Ingeniero en software	https://fisei.uta.edu.ec/v4.0/
4	Escuela Politécnica Superior del Litoral	Ingeniero/a en Ciencias de la Computación	https://www.espol.edu.ec/
5	Universidad Agraria del Ecuador	Ingeniero/a Computación e informática	http://www.uagraria.edu.ec/pregrado.html
6	Universidad de Cuenca	Ingeniería en Sistemas	https://www.ucuenca.edu.ec/oferta-academica/pregrado
7	Universidad Técnica de Babahoyo	Ingeniería en Sistemas	http://fafi.utb.edu.ec/sistemas-y-tecnologias
8	Universidad Técnica de Cotopaxi	Ingeniero en sistemas de la información	https://www.utc.edu.ec/

9	Universidad Técnica de Manabí	Ingeniero en sistemas de la información Ingeniero en sistemas de la información	https://www.utm.edu.ec/
10	Universidad de las Fuerzas Armadas	Ingeniero en software Ingeniero en tecnologías de la computación	https://www.espe.edu.ec/carreras/modalidad-presencial/ingenieria-de-software/
11	Universidad Técnica del Norte	Ingeniero en software	https://software.utn.edu.ec/
12	Universidad Técnica Estatal de Quevedo	Ingeniero en software	https://www.uteq.edu.ec/carrera/Software-7/
13	Universidad técnica Luis Vargas Torres de Esmeraldas	Ingeniero en software	https://www.uteq.edu.ec/carrera/Software-7/
14	Universidad Estatal del Sur de Manabí	Ingeniero en software	http://unesum.edu.ec/
15	Universidad Estatal Península de Santa Elena	Ingeniero en software	https://www.upse.edu.ec/
16	Universidad Laica Eloy Alfaro de Manabí	Ingeniería en sistemas	https://carreras.uleam.edu.ec/facci/comision-de-carrera/ingenieria-en-sistemas/
17	Universidad Tecnológica Equinoccial UTE	Ingeniero en ciencias de la computación	https://www.ute.edu.ec/
18	Universidad del Azuay	Ingeniero en ciencias de la computación	https://www.uazuay.edu.ec/
19	Universidad San Gregorio de Portoviejo	Ingeniero en ciencias de la computación	https://sangregorio.edu.ec/

20	Universidad Técnica Particular de Loja UTPL	Ingeniero en ciencias de la computación	https://www.utpl.edu.ec/
21	Universidad Internacional de Ecuador UIDE	Ingeniero en tecnologías de la información	https://www.uide.edu.ec/
		Ingeniero en Informática y Multimedia	https://www.uide.edu.ec/
22	Universidad de Otavalo	Ingeniero en tecnologías de la información	https://www.uotavalo.edu.ec/
23	Universidad Particular internacional SEK UISEK	Ingeniero en sistemas de la información	https://uisek.edu.ec/
24	Universidad del Pacífico escuela de negocios	Ingeniero en sistemas de la información	http://www.upacifico.edu.ec/website/index.php
25	Universidad Católica del Ecuador	Ingeniero en tecnologías de la información	https://www.puce.edu.ec/
26	Universidad Católica del Ecuador	Ingeniero de software	https://www.ucacue.edu.ec/
27	Universidad San francisco de Quito	Ingeniero en ciencias de la computación	https://www.usfq.edu.ec/es
28	Universidad del Río	Ingeniero de software	http://www.udr.edu.ec/
29	Universidad Tecnológica Indoamérica	Ingeniero en ciencias de la computación	https://uti.edu.ec/~utiweb/
30	Universidad Tecnológica ECOTEC	Ingeniero de software	https://www.ecotec.edu.ec/
31	Universidad Tecnológica empresarial de Guayaquil	Ingeniero de software	https://www.uteg.edu.ec/
32	Universidad Tecnológica Israel	Ingeniero en sistemas de la información	https://uisrael.edu.ec/
33	Universidad Autónoma de los Andes	Ingeniero de software	https://uniandes.edu.ec/

34	Universidad Metropolitana	Ingeniero en sistemas de la información	https://www.umet.edu.ec /
35	Universidad Iberoamericana del Ecuador	Ingeniero de software	https://www.unibe.edu.ec/
36	Universidad particular de especialidades espíritu santo	Ingeniero en ciencias de la computación	https://www.uees.edu.ec /
37	Universidad de los Hemisferios	Ingeniero en tecnologías de la información	https://www.uhemisferios.edu.ec/
38	Universidad de las Américas	Ingeniero en tecnologías de la información	https://www.uees.edu.ec /
39	Universidad Nacional de Chimborazo	Ingeniero en tecnologías de la información	https://www.unach.edu.ec/
40	Universidad Politécnica estatal del Carchi	Ingeniero en ciencias de la computación	https://www.upec.edu.ec /
41	Universidad Nacional de Loja	Ingeniero en ciencias de la computación	https://www.unl.edu.ec/

Nota. Instituciones de Educación Superior que ofertan Ciencias de la Computación, Informática y afines.

Si bien la educación en informática en el Ecuador ha avanzado en los últimos años, aún enfrenta retos y desafíos. Aquí se mencionan algunos de ellos:

1. La falta de recursos técnicos adecuados es uno de los mayores desafíos en la enseñanza de las ciencias de la computación. Muchas escuelas e instituciones educativas no cuentan con laboratorios de computación ni con los equipos e instalaciones necesarios. Esto dificulta la enseñanza de la informática y limita la capacidad de los estudiantes para adquirir experiencia en el uso de la tecnología, en especial de la programación.

2. La rápida evolución de la tecnología dificulta mantener actualizados los conocimientos y habilidades de los profesores. Esto puede afectar la calidad de la enseñanza

de la informática y limitar las oportunidades de los estudiantes para adquirir habilidades relevantes necesarias en el mercado laboral. Los planes de estudio en informática pueden no estar actualizados, en relación con los avances tecnológicos y las demandas del mercado laboral; esto ocasiona que la enseñanza de tecnologías y conceptos sea obsoleta, en consecuencia, no se prepara adecuadamente a los estudiantes para las necesidades del sector tecnológico, en especial en el desarrollo de software.

3. Aunque se han realizado esfuerzos para mejorar la conectividad y el acceso a la tecnología en Ecuador, todavía existe una brecha digital significativa, especialmente en áreas rurales y comunidades desfavorecidas. Esto limita el acceso de los estudiantes a recursos en línea, recursos informáticos y oportunidades de aprendizaje asistido por computadora.

4. En algunos casos, las ciencias de la computación se enseñan como una materia separada en lugar de integrarse en el plan de estudios. Esto puede limitar la comprensión de los estudiantes de cómo la informática se relaciona con otros campos de estudio y cómo se puede aplicar en todas las disciplinas.

A menudo hay una falta de orientación y asesoramiento profesional en materia de informática. Es posible que los estudiantes no estén al tanto de las diversas oportunidades y carreras disponibles en el campo, lo que puede afectar su interés y motivación para seguir una educación en informática. Estos desafíos y problemas requieren una atención continua de las instituciones educativas, el gobierno y otros actores relevantes. Es necesario invertir en infraestructura tecnológica, capacitar y reclutar docentes calificados, actualizar los planes de estudio, promover el acceso a la tecnología y fomentar la orientación profesional para superar estas limitaciones y mejorar la enseñanza de la informática en Ecuador.

5. Existen planes y programas de estudio específicos para la enseñanza de la informática en Ecuador. Estos planes se han actualizado para reflejar los avances tecnológicos y las demandas del mercado laboral. Se enfocan en áreas como

programación, bases de datos, redes de computadoras, seguridad informática y desarrollo de software, entre otros temas relevantes.

Ecuador cuenta con diversas instituciones educativas, tanto públicas como privadas, que ofrecen programas de formación en informática. Desde escuelas técnicas hasta universidades, los estudiantes pueden adquirir habilidades y conocimientos informáticos. Algunas universidades e institutos de educación superior también ofrecen títulos profesionales en informática, ingeniería de sistemas y tecnología de la información.

Además de la educación formal, Ecuador también ha introducido computadoras y certificaciones informáticas. Estos programas ofrecen, a profesionales y estudiantes, la oportunidad de adquirir habilidades especializadas en áreas como programación, análisis de datos, ciberseguridad y más.

4.2. Instituciones de educación superior participantes en los casos cuasiexperimentales

Para el desarrollo de la presente tesis doctoral se trabajó con tres IES, situadas en Ecuador, provincia de Loja, cantón Loja, donde se ofertan las carreras en el área de las ciencias de la computación e informática. A continuación, se detallan las características de las IES.

4.2.1. Universidad Técnica Particular de Loja

La Universidad Técnica Particular de Loja (UTPL) fue fundada por la Asociación Marista Ecuatoriana (AME) el 3 de mayo de 1971 y reconocida por el Estado ecuatoriano bajo Decreto Ejecutivo 646, publicado en el registro oficial N.º217 del 5 de mayo de 1971, se constituye como persona jurídica autónoma bajo el amparo del “Modus Vivendi” celebrado entre la Santa Sede y el Ecuador, teniendo en cuenta las normas de la Iglesia en su organización y gobierno.

El mentor y primer canciller de la Universidad fue el hermano marista Santiago Fernández García, y el primer rector de la UTPL fue el Ingeniero Alejo Valdivieso, posesionado el 3 de octubre de 1971. La UTPL es una institución autónoma, con finalidad social y pública, imparte enseñanza, desarrolla investigaciones con libertad científico-administrativa, y participa en planes de desarrollo del país (Universidad Técnica Particular de Loja [UTPL], 2021).

La UTPL es pionera en la modalidad de educación a distancia desde el año 1976, abriendo la posibilidad para acceder a una educación superior de calidad. En 1997 la Diócesis de Loja traspasa al Instituto Id de Cristo Redentor, Misioneros y Misioneras Identes la conducción de la Universidad, para que la dirija con total autonomía y en consonancia con el carisma Idente. La vida dentro de la UTPL se encamina a llevar a plenitud los ideales que dan origen a la Universidad, teniendo como visión el *"Humanismo de Cristo"* (UTPL, 2021).

Visión: La visión de la Universidad Técnica Particular de Loja es el humanismo de Cristo, que se traduce en sentido de perfección, en compromiso institucional, en servicio a la sociedad, en mejora continua y en la búsqueda constante de la excelencia.

El Humanismo de Cristo que, en su manifestación histórica y el desarrollo de su pensamiento en la tradición de la Iglesia Católica, propugna una universidad potenciadora, conforme a la dignidad que el ser humano tiene como *"hijo de Dios"* y que hace a la universidad acoger, defender y promover en la sociedad, el producto y la reflexión de toda experiencia humana (UTPL, 2021).

Misión: Desde la visión del humanismo de Cristo, la Universidad Técnica Particular de Loja tiene como misión: "Buscar la verdad y formar personas, a través de la ciencia, para servir a la sociedad" (UTPL, 2021).

La misión institucional armoniza la relación entre academia, investigación y vinculación, apuntando la centralidad de la búsqueda de la verdad y el servicio a la sociedad. La verdad como horizonte hacia el que dirigir, en comunión y respeto, las dimensiones

cognoscitivas, activas y vitales; una formación integral que aúne las dimensiones científico-técnicas de alta calidad, con las humanísticas, éticas y espirituales; un espíritu de investigación que contribuya al desarrollo de las ciencias experimentales y experienciales, y una disposición de servicio a la sociedad que suponga un efectivo aporte al desarrollo humanamente sustentable de su entorno local, del Ecuador y de toda la humanidad, con preferencia hacia los sectores menos favorecidos, todo ello desde el sentido que aporta la reflexión metafísica y la pedagogía Idente (UTPL, 2021).

Dentro de la oferta académica de la Universidad Técnica Particular de Loja hay 43 carreras de pregrado, 16 de maestrías de manera presencial; 23 carreras Modalidad a Distancia (UTPL, 2021).

En este contexto, en la Facultad de Ingenierías y Arquitectura se ofertan las Carreras de Computación y Tecnologías de la Información; pero por la naturaleza y tipo de modalidad se trabajó únicamente con Computación, que es carrera de modalidad presencial.

Descripción de la Carrera de Computación UTPL

En la tabla 8 se presentan las características de la Carrera de Computación de la UTPL.

Tabla 8

Características de la Carrera de Computación de la UTPL

Formar de manera integral a profesionales capaces de utilizar la tecnología informática de punta y los conocimientos teórico-prácticos para solucionar problemas relacionados con la ingeniería en ciencias computacionales. Emplear metodologías de enseñanza que priorizan las actividades prácticas, el fortalecimiento de las habilidades y destrezas propias de la profesión. Desarrollar la capacidad de actitud emprendedora, la formulación y gestión de proyectos, la investigación y resolución de problemas, y el razonamiento proactivo, generando

respuestas innovadoras a las necesidades y problemáticas del entorno laboral tan diverso y complejo para los expertos en ciencias computacionales (UTPL, 2021)¹.

Dentro de su malla curricular se dictan 7 asignaturas de programación; se complementa la formación profesional con interinarios académicos especializados.

Dentro del mismo contexto es preciso mencionar algunos hitos importantes de la UTPL a nivel internacional, nacional y local. Destacan los presentados en la tabla 9.

Tabla 9
Hitos de la UTPL

Hito	Descripción
Líder y pionera en Educación a Distancia en Latinoamérica	En 1976, la UTPL ya marcaba un hito histórico en el acceso a la educación superior en el país, convirtiéndose en la pionera dentro de Latinoamérica, en ofertar la carrera de Ciencias de la Educación, bajo la Modalidad de Educación a Distancia. Hoy, gracias al desarrollo de un modelo de educación basado en estándares vanguardistas y de calidad, cuenta con un entorno virtual de aprendizaje que permite al estudiante aprender de acuerdo con su disponibilidad de tiempo y que está activo las 24 horas del día; además de una plataforma virtual para acceder a material bibliográfico, recursos gráficos y audiovisuales, foros, chats, entre otros.
Investigaciones a nivel nacional e internacional	La UTPL destaca por su potencial investigativo, pues posee más de 2201 artículos científicos que han sido indexados en bases de datos de prestigio a nivel nacional e internacional, los cuales han sido publicados por 1397 autores afiliados a la UTPL, y son considerados como referentes de la investigación.

1 *Nota.* Descripción de las características de la Carrera de Computación de la UTPL adaptado de Universidad Técnica Particular de Loja.

Más de 151 proyectos de vinculación e innovaciones sociales	Gracias al apoyo de más de 100 instituciones públicas y privadas en conjunto con la comunidad UTPL, en los últimos 4 años se pudo crear y ejecutar 151 proyectos de servicio comunitario para mejorar la calidad de vida de la sociedad, en estas iniciativas participaron 11998 estudiantes y 764 docentes UTPL, quienes contribuyeron al cumplimiento de los Objetivos de Desarrollo Sostenible de la ONU.
Implementación de herramientas tecnológicas de realidad aumentada	Para potenciar los conocimientos de todas las carreras, pero en especial de Biología, Ciencias de la Educación, Derecho y Medicina, que están enmarcadas en una metodología de enseñanza basada en proyectos, problemas o casos, fue necesario implementar tecnologías de realidad aumentada. Estos espacios permitieron que los estudiantes de la Modalidad Presencial y a Distancia aprendan de forma virtual situaciones reales como autopsias, anatomía o estadística. Se desarrollaron mundos virtuales, donde cada estudiante diseña su avatar para explorar áreas de práctica como en un set de televisión, una sala de audiencia, realizar juegos de roles o hacer visitas guiadas en turismo.
Centros UTPL en el país y el extranjero	Hasta la actualidad la universidad ha crecido, cuenta con más de 80 centros en Ecuador y es la única universidad ecuatoriana con presencia en Europa y Norteamérica, con base a sus líneas estratégicas de desarrollo de una Universidad como Alma Mater para el siglo XXI, la investigación que fomenta el desarrollo local y nacional, la docencia de alto nivel con profesores capacitados en brindar una educación pedagógica eficaz y el liderazgo que por medio de la innovación permite la mejora continua. Actualmente, la UTPL cuenta con más de 39 mil estudiantes en la Modalidad Abierta y a Distancia.

The Impact Ranking 2021	La UTPL se ubica en THE IMPACT RANKING 2021 en el tercer lugar a nivel nacional y entre las 300 a 400 universidades a nivel mundial.
Scimago Institutions Rankings 2020	La UTPL en el referente Scimago Institutions Rankings 2020 a la Evaluación científica de instituciones de educación superior centradas en investigación.
Impacto Social	Tercer lugar en impacto social
Impacto en la investigación	Quinto lugar en investigación
A nivel nacional	Décimo primer lugar en innovación
Posición en Latinoamericana.	Centésimo vigésimo noveno lugar a nivel latinoamericano. ²

La capacidad y calidad de sus docentes, así como el robusto sistema educativo que guarda la metodología de estudio de la UTPL, le ha permitido que con la llegada de la pandemia no solo se puedan impartir clases basadas en el desarrollo de pedagogías y tecnologías actualizadas que instruyan, motiven y acompañen al estudiante, sino también que puedan enfrentar la virtualidad de una educación de calidad y con propósito. Por estas razones, la universidad celebró más de 51 años de vida institucional con más compromiso y visión a largo plazo (UTPL, 2021).

4.2.2. Universidad Internacional del Ecuador

La Universidad Internacional del Ecuador (UIDE) debe su existencia a la visión de Jorge Fernández Salazar. El 21 de octubre de 1992, con la aprobación del Consejo Nacional de Universidades y Escuelas Politécnicas (CONUEP), Marcelo Fernández Sánchez, Canciller Fundador, materializó la visión de su padre iniciando las actividades de la UIDE (Universidad Internacional del Ecuador [UIDE], 2021).

² Nota. Hitos alcanzados por la UTPL adaptada de (Universidad Técnica Particular de Loja, 2021).

Misión: Brindar una educación de calidad para una vida exitosa.

Visión: Ser una de las mejores universidades de América Latina para el año 2035 y contribuir a la integración del continente americano. Dentro de la formación profesional en el área técnica, la UIDE oferta la Carrera de Ingeniería Informática y Multimedia en su extensión Loja, en la cual se desarrollaron dos de los casos cuasiexperimentales.

Dentro del contexto de la Carrera de Informática y Multimedia de la UIDE es importante destacar algunas características. Se resumen en la tabla 10.

Tabla 10

Descripción del perfil de la carrera de Informática y Multimedia

El Ingeniero en Informática y Multimedia podrá diseñar y construir soluciones de informática y multimedia a través del uso de herramientas actuales; así mismo podrá planear y desarrollar proyectos innovadores con transferencia tecnológica, investigación y uso adecuado de las TIC.

Desarrollar investigación que genere soluciones en sistemas de información, multimedia, telecomunicaciones y hardware.

Comunicar fácilmente soluciones tecnológicas integrales a equipos multidisciplinarios por su facilidad y conocimiento de inglés, que solo la UIDE entrega.³

En la tabla 11 se describe los hitos de la UIDE.

Tabla 11

Hitos de la UIDE

1996: El 24 de julio el Congreso Nacional aprobó la ley de creación de la Universidad Internacional del Ecuador con publicación en el Registro Oficial No. 15.

1997: La UIDE es registrada por el Consejo Nacional de Enseñanza Superior Universitaria Privada (CONESUP), mediante oficio No. 144-P.

2002: Se crea la Facultad de Ciencias Médicas que actualmente es una de las más prestigiosas del país.

Se aprueba la creación de la sede Loja mediante resolución No. CRP-57-082-02.

³ NOTA: CARACTERÍSTICAS DEL PERFIL DEL INGENIERO EN INFORMÁTICA Y MULTIMEDIA ADAPTADO DE UIDE, 2021.

2004: La UIDE se traslada al campus principal, en donde funciona actualmente, ubicado en la avenida Jorge Fernández y Simón Bolívar.

2010: El 14 de octubre inician las actividades del Consultorio Jurídico Gratuito de la UIDE con autorización del Ministerio de Justicia y Derechos Humanos.

2014: En noviembre, la Escuela de Medicina alcanza el 100 % de aprobación en el examen de habilitación profesional del Consejo de Evaluación Acreditación y Aseguramiento de la Calidad de la Educación Superior (CEAACES).

2015: El CEAACES aprueba el funcionamiento de las extensiones de Guayaquil y Loja de la UIDE. La Facultad de Medicina recibe la acreditación incondicional del CEAACES. El Consultorio Jurídico recibe la acreditación anual de funcionamiento por parte de la Defensoría Pública

2016: La carrera de Odontología recibe la acreditación por parte del CEAACES. La UIDE fue la única universidad privada, autofinanciada y con extensiones en ser recategorizada entre las mejores universidades del Ecuador.

2021: La UIDE se une a Cintana Alliance y anuncia su afiliación de clase mundial a Arizona State University (ASU). La UIDE se convierte en la única universidad del Ecuador, de entre más de 60 instituciones de educación superior, en ser escogida por una casa de estudios estadounidense para formar una alianza estratégica en nuestro país.⁴

4.2.3. Universidad Nacional de Loja

La Universidad Nacional de Loja (UNL) es un Centro de Educación Superior, de alta calidad académica y humanística, que ofrece formación en los niveles técnico y tecnológico superior, profesional o de tercer nivel y de postgrado o cuarto nivel. Realiza investigación científico-técnica sobre los problemas del entorno, con calidad, pertinencia y equidad, a fin de coadyuvar al desarrollo sustentable de la región y del país, interactuando con la comunidad, generando propuestas alternativas a los problemas nacionales con responsabilidad social, reconociendo y promoviendo la diversidad cultural y étnica la

⁴ Nota: Hitos de la UIDE Recuperado: (UIDE,2021 <https://www.uide.edu.ec/historia/>)

sabiduría popular, apoyándose en el avance científico y tecnológico, en procura de mejorar la calidad de vida del pueblo ecuatoriano, que tiene entre sus fines promover la cooperación interinstitucional y el intercambio académico, técnico y sociocultural con gobiernos locales, gobiernos provinciales, instituciones públicas y demás organismos nacionales e internacionales.

Desde el mes de septiembre del 2008, la UNL está formando profesionales en Administración Pública, cuya vocación es el servicio a la colectividad, ejerciendo un liderazgo social, desde el punto de vista del desarrollo humano sustentable. La sede principal de la UNL está ubicada en el barrio La Argelia, ciudadela Guillermo Falconí Espinosa.

Fundada el 09 de octubre de 1943, mediante Decreto Oficial expedido por el presidente de la República del Ecuador, Dr. Carlos Arroyo del Río, según el Estatuto Orgánico, la representación legal de la Universidad Nacional de Loja le corresponde al señor rector.

La UNL a es una Institución de Educación Superior, laica, autónoma, de derecho público, con personería jurídica y sin fines de lucro, de alta calidad académica y humanística, que ofrece formación en los niveles técnico y tecnológico superior, profesional o de tercer nivel y de postgrado o cuarto nivel; realiza investigación científico-técnica sobre los problemas del entorno, con calidad, pertinencia y equidad, a fin de coadyuvar al desarrollo sustentable de la región y del país, interactuando con la comunidad, generando propuestas alternativas a los problemas nacionales, con responsabilidad social; reconociendo y promoviendo la diversidad cultural y étnica y la sabiduría popular, apoyándose en el avance científico y tecnológico, en procura de mejorar la calidad de vida del pueblo ecuatoriano (Universidad Nacional de Loja [UNL],2021) .

Tiene como misión la formación académica y profesional, con sólidas bases científicas y técnicas, pertinencia social y valores; la generación y aplicación de conocimientos científicos, tecnológicos y técnicos, que aporten al desarrollo integral del entorno y al avance

de la ciencia; el fortalecimiento del pensamiento, la promoción, desarrollo y difusión de los saberes y culturas; y la prestación de servicios especializados (Universidad Nacional de Loja, 2021).

Dentro del contexto de formación en el área técnica, la UNL oferta la carrera de Computación.

La carrera de Computación analiza los fundamentos científicos para desarrollar soluciones en el contexto de sistemas inteligentes, ingeniería de software, computación aplicada, con el propósito de incursionar en la industria o en el liderazgo de proyectos de investigación formativa y/o investigación científica/tecnológica, bajo los principios de calidad, planificación, gestión, desarrollo, responsabilidad, honestidad e interculturalidad (UNL, 2021). Busca formar profesionales con bases sólidas y enfoques científico-humanistas en los niveles tecnológico-técnico-artesanal y de pregrado, acorde a la vanguardia tecnológica en los avances científicos (UNL, 2021).

La universidad se propone sistematizar los conocimientos científico-técnicos universales a través de la investigación para confrontar su aplicación a la realidad regional y nacional, mediante la generación de conocimientos y soluciones innovadoras para enfrentar los problemas del entorno con la participación de los actores sociales involucrados (UNL, 2021).

Su propósito es aportar efectivamente al desarrollo del país y en particular de la Región Sur, a través de la formación de talentos humanos, generación de conocimientos y vinculación con la colectividad, con la visión de aprovechar los recursos naturales no renovables con criterio técnico, equidad social, rentabilidad económica y sostenibilidad ambiental; para solidificar industrias electromecánicas y de la construcción que generen trabajo productivo, utilizando los recursos propios incorporando la informática como herramienta que sustente su accionar (UNL, 2021).

En la tabla 12 se presentan los principales hitos de la UNL.

Tabla 12

Hitos de la Universidad Nacional de Loja

Escenarios de aprendizaje, investigación y vinculación

La UNL tiene sus centros de investigación repotenciados donde sus estudiantes y docentes generan aportes científicos para el país y el mundo a través del Centro de Biotecnología, Centro de Investigaciones (CITIAB), Centro de Investigaciones Sociales y Económicas (CISE), Centro de Investigaciones Territoriales (CIT) y Laboratorio de Análisis Químico.

Otros escenarios de investigación son las estaciones experimentales: El Padmi, Zapotepamba, La Argelia, Punzara, El Chilco, Jardín Botánico y Parque Universitario. Al igual que los laboratorios de las carreras fueron intervenidos y dotados de insumos y tecnología para lograr sus fines.

Acreditación

Luego de siete años y de haber realizado un trabajo entre toda la comunidad universitaria, la UNL fue acreditada para los próximos cinco años por el Consejo de Aseguramiento de la Calidad de la Educación Superior (CACES).

“La mejor forma de rendirle tributo a nuestra casa de estudios superiores en su aniversario de fundación es con su acreditación”, comentó el rector Nikolay Aguirre. A la vez señaló que este triunfo institucional les pertenece a los estudiantes, docentes, administrativos, trabajadores y al pueblo de Loja. “Hoy tenemos una universidad con prestigio, reconocida y transformada”, puntualizó.

Credibilidad

La confianza se evidencia en los 12691 estudiantes que se forman en el Alma Mater lojana. Para Marco León, estudiante de Comunicación Social, la evolución de la UNL es notoria, “vemos una universidad conectada al mundo, moderna y que investiga”. Johanna Bermeo, alumna de la misma carrera dice que ahora estudia en una universidad acreditada y de prestigio. Mientras que, para el universitario de Psicología Clínica, Carlos

Daniel Hidalgo, observa una institución organizada e incluyente donde participan todos de su transformación.⁵

La UNL se ha destacado por su compromiso con la excelencia académica y la investigación científica. Cuenta con laboratorios, centros de investigación y programas de posgrado que fomentan la generación de conocimiento y la innovación. Además de su enfoque académico, la UNL tiene una fuerte presencia en la comunidad a través de proyectos de extensión y servicio a la sociedad. También ha establecido convenios de cooperación con otras instituciones nacionales e internacionales para promover el intercambio académico y la colaboración en investigación.

⁵ Nota: Hitos de la UNL, adaptado, <https://unl.edu.ec/noticia/unl-cumple-161-anos>

Capítulo cinco: Metodología de la investigación

5.1 Método de investigación Diseño de la Ciencia

El DC es un método de investigación iterativo basado en el paradigma de resolución de problemas que incluye el diseño, creación y validación de artefactos para resolver un problema identificado (Hevner, 2007). En nuestro caso se centró en la CN y CA, en el proceso de desarrollo de destrezas tecnológicas, enfocadas a la resolución de problemas de aprendizaje de codificación y paradigmas computacionales en estudiantes de Ingeniería Informática y ciencias de la computación.

En el mismo orden de ideas, DC es tanto un proceso como un producto. Un proceso que define una secuencia de actividades para obtener un producto artefacto. El conocimiento y la comprensión de un dominio del problema y su solución se lograron en la construcción y aplicación del artefacto diseñado. DC creó y evaluó los artefactos de TI, destinados a resolver los problemas organizativos identificados en el contexto de la CN y CA. Dichos artefactos fueron representados en una forma estructurada que puede variar desde software, lógica formal y matemáticas rigurosas hasta descripciones de lenguaje natural informal.

De Benito y Salinas (2016) sostiene que la investigación basada en el DC es un tipo de investigación dirigida hacia la innovación educativa, cuya característica fundamental consiste en la introducción de un elemento nuevo para transformar una situación; que en nuestro contexto fue en la propuesta de un modelo pedagógico basado en la CN y CA para el desarrollo de las habilidades de programación (p.45).

5.1.1 Implementación de la metodología propuesta Diseño de la Ciencia

En la presente memoria se puso en marcha un estudio reflexivo para probar y refinar entornos de aprendizajes innovadores, así como para definir nuevos principios de diseño bajo la metodología descrita previamente. Esto implica varios pasos clave llevados a cabo en esta investigación:

1. Definición de objetivos: Se determinó qué aspectos del entorno de aprendizaje se probó e innovó, esto incluyó métodos pedagógicos, tecnología educativa CN y CA, estructuras físicas o virtuales y dinámicas de interacción.
2. Diseño del estudio: Se desarrolló un diseño de investigación adecuado para los objetivos. Para la elección de estos diseños se consideraron las preguntas de investigación específicas y el grado de control sobre las variables involucradas. También es importante considerar las implicaciones éticas y prácticas de cada diseño.
3. Selección de participantes: Se definió y reclutó seis grupos de participantes de tres IES Ecuatorianas, que representan adecuadamente la población de interés. Esto fue crucial para asegurar los resultados.

Definición de la población objetivo: Antes de reclutar participantes, fue esencial definir claramente la población de interés. Se hizo un análisis de las características demográficas, geográficas, socioculturales y otros criterios relevantes para la investigación propuesta. En la tabla 13 se presenta una síntesis de los casos cuasiexperimentales desarrollados.

Tabla 13
Síntesis de los casos cuasiexperimentales

IES	Asignaturas	Estudiantes	Horas	Herramientas Utilizadas
Universidad Técnica Particular de Loja (UTPL)	Programación Avanzada	25	128	Egroupware, REDMINE y Open Atrium Git Hub WordPress
	Programación Web	25	128	Spring boot, Visual Code Tablero Kanban de Tobeio
Universidad Nacional de Loja (UNL)	Proyectos Tecnológicos I (Enfoques de Programación)	20	120	Project Open, Jira, StormBoard.
	Computación en la Nube	29	128	GitHub, TeamBOX, DOOPHY
	Programación	15	128	AWS Cloud9, Diggo y GitHub.

Universidad Internacional del Ecuador (UIDE)	Cliente Servidor (Programación Web y Móvil) Programación Orientada a Objetos	25	128	Alfresco, DotProject, Redbooth
Total, de estudiantes		139		

4. Recolección de datos: Recoger datos cuantitativos y cualitativos para evaluar la eficacia del entorno de aprendizaje. Esto abarca pruebas de rendimiento, encuestas de satisfacción, entrevistas, observaciones, entre otros.
5. Refinamiento y reiteración: Basado en los resultados, hace ajustes y refinamientos al diseño del entorno de aprendizaje. Este es un proceso iterativo que puede requerir varias rondas de pruebas y ajustes.
6. Documentación y difusión: Documentar el proceso y los resultados del estudio, y compartirlos con la comunidad educativa para fomentar el conocimiento y la adopción de prácticas efectivas.

Una característica clave de la investigación en DC como método es que se lo orientó a la resolución de problemas específicos para obtener una solución satisfactoria a la situación, incluso, si la solución no es óptima. Sin embargo, en las soluciones generadas por la ciencia del diseño, la investigación, debe ser susceptible de generalización para una clase específica de problemas (van Aken, 2004, 2005; Sein et al., 2011; Vaishnavi y Kuechler, 2011).

A continuación, se detallan las características de cada fase DC desarrollada en la presente tesis doctoral.

1. Identificación y motivación del problema. La investigación desarrollada con el método de investigación basada en el DC produjo artefactos viables en forma de construcción, modelo, método o instanciación. En esta etapa, se justificó la relevancia de la

investigación, considerando su pertinencia, la importancia del problema y la aplicabilidad de la solución propuesta (Peffer et al., 2007).

La motivación del problema surgió de la necesidad de abordar una brecha en la práctica y en la teoría en las ciencias de la computación e informática, donde las soluciones existentes son insuficientes o inadecuadas en el contexto del desarrollo de las habilidades de la programación.

2. Relevancia del problema. El propósito de la investigación en ciencias del diseño fue desarrollar soluciones para resolver problemas relevantes para las organizaciones. Peffer et al.(2007) sugieren que los objetivos esperados al resolver el problema pueden ser tanto cuantitativos como cualitativos.

Lo que se busca en el modelo propuesto:

- Mejorar la creatividad del estudiantado.
 - Desarrollar la autonomía.
 - Promover la autoformación.
 - Proveer el aprendizaje activo.
 - Proveer el aprendizaje social.
 - Generar encuentros comunicativos.
 - Provocar la reflexión entre los actores educativos.
 - Mejorar la interacción entre docente y estudiantes.
 - Evaluar el aprendizaje en contextos reales mediante la fase cuasi experimental.
3. Diseño y desarrollo. En esta etapa se desarrollaron los diferentes artefactos que ayudaron a resolver el problema. La utilidad, la calidad y la eficacia del artefacto se demostraron rigurosamente mediante métodos de evaluación bien ejecutados. En la etapa de diseño y desarrollo, las funcionalidades deseadas para el artefacto, su arquitectura propuesta y su desarrollo fueron definidas. Se utilizaron los conocimientos teóricos existentes.

Durante la etapa de diseño y desarrollo:

1. Definición de funcionalidades: Se especificaron las funciones principales que el artefacto debe cumplir para abordar el problema identificado.
2. Arquitectura propuesta: Se diseñó una estructura detallada del artefacto, indicando cómo se relacionan sus componentes y cómo funcionarán en conjunto.
3. Desarrollo: Se construyó el artefacto siguiendo el diseño propuesto, integrando los componentes necesarios para cumplir con las funcionalidades.
4. Evaluación: La utilidad, calidad y eficacia del artefacto se validaron con métodos de evaluación adecuados, como pruebas funcionales, revisiones de calidad o retroalimentación de usuarios. Esto aseguró que el artefacto cumpla con los objetivos iniciales y resuelva el problema de manera efectiva.

Se aplicaron conocimientos teóricos para fundamentar el diseño, lo que anticipó desafíos y encaminó el desarrollo hacia un enfoque bien fundamentado. En cada caso cuasi experimental se generaron diferentes artefactos (aplicaciones de escritorio, web, móviles) en función de la planificación de cada asignatura.

4. Evaluación. La investigación realizada por el método de investigación de la ciencia del diseño proporcionó contribuciones claras y verificables en las áreas específicas de los artefactos desarrollados y presentó una base clara sobre los fundamentos del diseño y/o las metodologías de diseño. El cuarto paso del método, recomendado por Peffers et al. (2007), se refiere a la demostración, es decir, uso del artefacto para resolver el problema. Este paso puede ser realizado mediante experimentación y simulación (Peffers et al., 2007).

El modelo propuesto aporta en el desarrollo de las habilidades de programación y habilidades blandas de los participantes. La integración de la programación con el desarrollo de habilidades blandas es crucial en el ambiente laboral actual. Las habilidades de programación son esenciales para resolver problemas técnicos, mientras que las habilidades blandas, como la comunicación efectiva, el trabajo en equipo y la gestión del

tiempo, son imprescindibles para trabajar eficazmente en proyectos colaborativos y para ascender en cualquier carrera en especial las ciencias de la computación e informática.

5. Demostración. La investigación se basó en la aplicación de métodos rigurosos, especialmente en la construcción de los artefactos. Se observó y midió el comportamiento del artefacto para resolver el problema (Peppers et al., 2007). En esta fase se compararon los resultados de rendimiento del artefacto con el requisito para la resolución del problema. Si el resultado no satisface las expectativas, el investigador puede volver al diseño para desarrollar un nuevo artefacto (Peppers et al., 2007).

En la fase descrita, se llevó a cabo la evaluación del artefacto desarrollado. Esta evaluación es crucial porque permitió verificar si el artefacto cumple con los requisitos necesarios para resolver el problema planteado. Se utilizaron métodos rigurosos para observar y medir el comportamiento del artefacto. Este proceso iterativo ayuda a refinar el artefacto hasta que se adecue satisfactoriamente a las necesidades del problema.

Este enfoque garantizó que los resultados de la investigación sean prácticos y aplicables en situaciones reales, permitiendo a los investigadores hacer contribuciones significativas en sus respectivos campos.

6. Comunicación. La búsqueda de un artefacto efectivo requiere el uso de los medios disponibles para lograr los propósitos deseados, mientras se satisfacen las leyes que rigen el entorno en el que se estudia el problema. La investigación realizada debe presentarse tanto a una audiencia más orientada a la tecnología como a una más orientada a la gestión. Los resultados generados de la presente tesis doctoral se presentan en la tabla 2.

Las disciplinas de ingeniería y educación aceptan el diseño de la ciencia como metodología de investigación válida y valiosa, porque la cultura de investigación en ingeniería coloca valor explícito en las soluciones de forma incremental, eficaces y aplicables a problemas. El principio fundamental del DC es que el conocimiento y la comprensión de un

problema de diseño y su solución se adquieren en la construcción y aplicación de un artefacto, en nuestro caso enfocado al campo educativo, en especial, en la adquisición de destrezas de programación.

La importancia del método de investigación en el DC radicó en su capacidad para asegurar la validez y confiabilidad de los resultados obtenidos. A través de la aplicación rigurosa del método de investigación, se logró minimizar los sesgos y errores en la investigación y se garantizó la reproducibilidad de los resultados. Además, el método de investigación posibilitó el avance del conocimiento científico al proporcionar una estructura para llevar a cabo estudios y experimentos. También, este método facilitó la comparación de resultados entre diferentes investigaciones que contribuyen al desarrollo de teorías científicas sólidas.

5.2 Revisión sistemática de la literatura

García-Peñalvo (2022) manifiesta que la revisión sistemática de literatura es un método para identificar, evaluar e interpretar el trabajo de académicos y profesionales en un campo elegido. Su propósito es identificar lagunas en el conocimiento y necesidades de investigación en un campo concreto. Las revisiones sistemáticas, de literatura, implicó identificar, sintetizar y evaluar toda la evidencia disponible, cuantitativa y/o cualitativa, con el fin de generar una respuesta robusta, empíricamente derivada a una pregunta de investigación (Kitchenham et al., 2002).

Según Kitchenham et al. (2002) la revisión sistemática de literatura se propuso en el Software Engineering Research como un método para reportar conclusiones confiables sobre un área de investigación, mientras se recolecta sistemáticamente información de calidad en nuestro contexto en la CN y CA como base para la propuesta de un modelo pedagógico. Dentro del proceso metodológico, la revisión sistemática de la literatura presenta las fases que se detallan en la tabla 14.

Tabla 14

Fases de la revisión sistemática de la literatura

Planificación de la revisión. Esta fase tuvo como objetivo desarrollar el protocolo de revisión, que define los métodos para llevar a cabo un SLR específico, reduciendo la posibilidad de que sea impulsado por las expectativas de la investigación.

Desarrollo de la revisión. Esta fase tuvo como objetivo ejecutar el protocolo anterior.

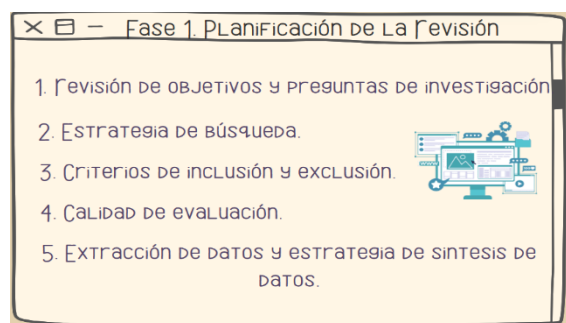
Informar la revisión. Esta fase tuvo como objetivo proporcionar los resultados obtenidos para la comunidad.

5.2.1 Planificación de la revisión

El propósito de este paso es definir los objetivos planteados en las preguntas de investigación y el protocolo de evaluación formal. En esta fase, se establecieron los pasos a seguir para implementar el protocolo. En la Figura 26 se presentan las fases de la planificación de la revisión sistemática de la literatura.

Figura 26

Planificación de la revisión sistemática de la literatura



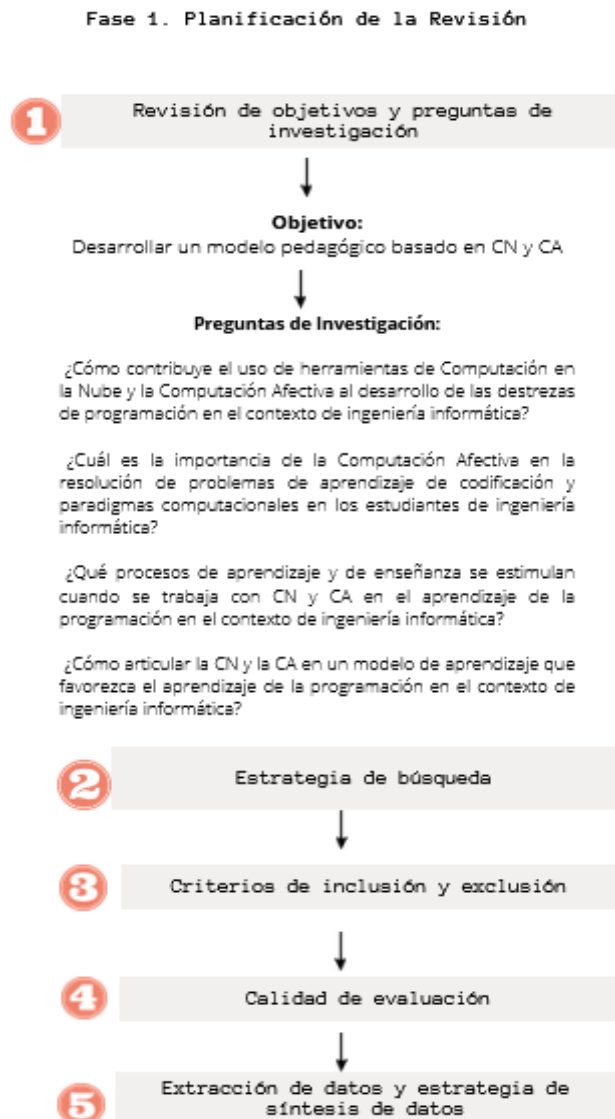
Nota. Fases de la planificación de la revisión sistemática de la literatura adaptada de Kitchenham (2002)

En esta fase se revisaron los objetivos y se buscaron los estudios primarios fundamentales, para establecer los elementos necesarios para el desarrollo de un modelo pedagógico basado en CN y CA para el desarrollo de las habilidades de programación, en los estudiantes de ciencias de la computación e ingeniería informática.

En la figura 27 se presenta la revisión de objetivos y preguntas de investigación.

Figura 27

Paso 1 de la planificación de la revisión



Nota. Descripción del paso 1 de la Planificación de la Revisión Sistemática de la Literatura adaptado de adaptada de Kitchenham (2002).

La primera estrategia de búsqueda, para planificar la revisión, fue identificar las bases de datos más relevantes y determinar los estudios primarios. Estas bases de datos más relevantes se muestran en las Tablas 15 y 16 donde, se ingresaron las cadenas de búsqueda.

Tabla 15
Cloud computing on teaching programming

Base de Datos	Artículos encontrados
Scopus	3766
IEEE Xplore Digital Library	139
ACM Digital Library	618788
Springer	16122
Web of Science	60
Science Direct	4725
ERIC	72
Total	643672

Nota. Base de Datos consultadas con primer criterio de búsqueda.

Tabla 16
Cloud computing and affective computing on teaching programming

Base de Datos	Artículos encontrados
Scopus	216
IEEE Xplore Digital Library	1
ACM Digital Library	619414
Springer	1821
Web of Science	1
Science Direct	2535
ERIC	17
Total	624005

Nota. Base de Datos consultadas con segundo criterio de búsqueda

En la Figura 28 se presenta el paso antes detallado.

Figura 28

Paso 2 de la planificación de la revisión sistemática de la literatura

Fase 1. Planificación de la Revisión



Nota. Paso 2 de la planificación de la revisión sistemática de la literatura adaptada de adaptada de Kitchenham (2002)

Una vez que se identificaron las bases de datos mencionadas anteriormente, se formaron las cadenas de búsqueda para determinar los estudios primarios. Se identificaron los documentos relevantes para responder las preguntas de investigación propuestas y se establecieron los criterios para los documentos de inclusión y exclusión referente al modelo pedagógico a desarrollar. Seguidamente se buscó una forma de extraer estos documentos. Se realizó una búsqueda automática en las bases de datos electrónicas científicas y se validó mediante el protocolo SLR (Kitchenham y Charters, 2007). Así mismo, se establecieron los criterios de inclusión y exclusión de los estudios primarios; para ello se usó la rúbrica

propuesta por (Dybå, y Dingsøyr, 2008). En la Figura 29 se presenta el proceso antes mencionado.

Figura 29
Criterios de inclusión y exclusión de los estudios primarios



Nota. Criterios de inclusión y exclusión de los estudios primarios adaptado de adaptada de Kitchenham (2002)

Seguidamente, se estableció la calidad de la evaluación de los estudios primarios; considerando si las contribuciones definen claramente los objetivos y tienen relación con la búsqueda propuesta. Así mismo, se analizó si las contribuciones manejan métodos empíricos para obtener evidencias. También, se analizó si el estudio describe el contexto de la CN y CA y su relación para el desarrollo de un modelo pedagógico que aporte al incremento de las habilidades de programación. Finalmente, se estableció el proceso para la extracción de datos y las respectivas estrategias de síntesis basadas en lo propuesto por Quezada et. al. (2020).

5.2.2 Desarrollo de la revisión

Una vez que la fase de planificación fue validada, se realizó la búsqueda de los estudios primarios en cada base de datos, para responder las preguntas de investigación. Además, el trabajo de investigación presentado en este documento consideró los criterios de inclusión y exclusión para identificar aquellos estudios primarios que proporcionaron evidencia directa para responder a las preguntas de investigación. Adicionalmente, para reducir la probabilidad de sesgo, los criterios de selección se aprobaron durante la definición del protocolo. Aquí, la selección de estudios primarios implicó un proceso de varias etapas. Inicialmente, los criterios de selección se interpretaron abiertamente, a continuación, estos criterios se refinaron para evitar la duplicación de datos.

La SLR requirió tomar en cuenta explícitamente algunos criterios de exclusión para evaluar la idoneidad de la información en cada estudio primario y responder a las preguntas de investigación (Kitchenham y Charters, 2007).

Criterios de inclusión (IC). Se incluyeron materiales científicos (es decir, documentos, informes de experiencias, resúmenes de talleres, etc.), escritos en inglés, español y accesibles digitalmente.

Criterios de exclusión (CE). Documentos referentes a CN y CA, pero no relacionados con educación.

- Documentos relacionados con la CN y CA, pero no relacionados con las preguntas de investigación de este estudio.

- Eliminar falsos positivos. Es decir, se eliminaron los documentos que no formaban parte de la investigación en la cadena de búsqueda.

- Se excluyeron los documentos en forma de resúmenes, tutoriales o charlas que no proporcionaron suficientes detalles para la revisión sistemática de la literatura.

- Malos argumentos. Es decir, se excluyeron los estudios con baja relevancia según la investigación.

- Reducción de absurdos, aquellos estudios que no cumplieron con los criterios fueron excluidos.

- Luego, como continuación del proceso de selección de PS, se aplicaron criterios de inclusión y exclusión sobre cuestiones prácticas.

Ejemplos de estos criterios fueron: idioma, revista, autores, participantes o sujetos, diseño de investigación, método de muestreo y fecha de publicación (Esler, 2018).

Para interpretar los hallazgos en PS, se determinó la importancia de la calidad científica de los artículos en estudio. Los criterios de inclusión y exclusión indicaron la credibilidad del PS analizado e identificaron posibles limitaciones de la búsqueda actual (Brereton et al., 2007; Keele, 2007). Los criterios de calidad fueron los siguientes: rigor, credibilidad y relevancia.

Para guiar la interpretación de los hallazgos en los estudios incluidos, y determinar la fuerza de las inferencias, evaluamos los criterios de calidad científica de los estudios seleccionados para determinar la relevancia de los resultados obtenidos en la revisión realizada. Estos criterios indican la credibilidad de un estudio individual al sintetizar resultados. El resultado de la evaluación de calidad de los estudios incluidos puede revelar las limitaciones potenciales de la búsqueda actual y guiar la investigación futura en el campo

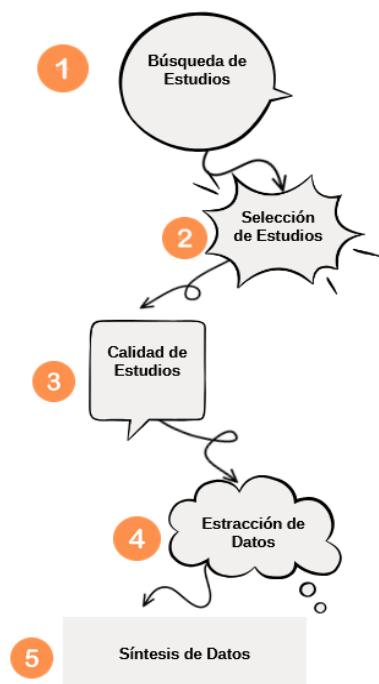
(Keele, 2007). Finalmente, como parte de esta fase, la revisión sistemática de la literatura la estrategia de extracción de datos permitió identificar la información requerida para responder el RQS.

Los resultados de las búsquedas automáticas y manuales, después de la aplicación de los criterios de selección, fueron aplicados al concepto de bola de nieve, es decir, las referencias bibliográficas de todos los artículos seleccionados se consideraron estudios potenciales para apoyar la investigación (Medeiros et al., 2019). En la Figura 30 se presentan los pasos antes mencionados.

Figura 30

Fases del desarrollo del protocolo de la revisión sistemática de la literatura

Fase 2. Conducción de la Revisión



Nota. Desarrollo del Protocolo de la Revisión Sistemática de la Literatura

5.2.3 Informar la revisión

Esta fase de la revisión sistemática de la literatura consta de los pasos presentados en la Figura 31.

Figura 31
Reporte de la revisión

Fase 3. Reporte de la Revisión



Primero se extrajo la información de los artículos. En este caso, se utilizaron hojas electrónicas, Mendeley y Atlas TI 9.0, como herramientas de software (Muñoz et al., 2011). Mendeley nos permitió organizar las referencias bibliográficas que son la base científica de esta revisión.

En esta revisión, Atlas TI permitió asociar códigos o etiquetas con fragmentos de texto (Hwang, 2007) de los estudios primarios y creación de las redes de conceptos que serán la base del modelo de descripción. Además, Atlas TI nos permitió buscar códigos de patrón, clasificarlos para establecer los elementos para describir el contexto de las herramientas de CN y CA.

En segundo lugar, la información obtenida se sintetizó para comprender el contexto de la CN y CA.

En tercer lugar, los formularios de extracción de datos se desarrollaron y cargaron en un repositorio en línea. En este punto, es importante mencionar que las estrategias de síntesis son importantes no solo porque incluyen múltiples publicaciones de los mismos datos en una

SLR, sino también porque evitan informes duplicados que podrían sesgar los resultados del estudio.

El informe de la fase de revisión fue respaldado por el análisis de calidad que se realizó para cada estudio primario. Para obtener el concepto de redes, se diseñó el ciclo iterativo. Luego, los conceptos de redes se agruparon en categorías. Después de eso, para cada categoría, se crearon interacciones y también se desarrollaron los esquemas de representación que son la base de lo propuesto en la presente tesis doctoral.

5.3 Casos Cuasi experimentales

El país de desarrollo de los casos experimentales fue el Ecuador, provincia de Loja, cantón Loja, ciudad Loja, en tres instituciones de educación superior (IES): Universidad Técnica Particular de Loja (UTPL), Universidad Internacional del Ecuador (UIDE), Universidad Nacional de Loja (UNL).

Las características más importantes de los casos cuasi experimentales se detallan en la tabla 17.

Tabla 17
Detalles de los casos cuasi experimentales

Caso	Detalles	Herramientas
Cuasi experimental		Utilizadas
UTPL 1	En el caso cuasi experimental uno, desarrollado en la UTPL, se trabajó con estudiantes que cursaron la asignatura de Programación Avanzada ubicada en el cuarto ciclo de la carrera de Sistemas Informáticos y Computación, con una carga horaria de 4 créditos, considerando que cada crédito equivale a 32 horas académicas.	WordPress con la integración de pluings de desarrollo y módulos de acuerdo con los requerimientos del proyecto desarrollo. Egroupware, REDMINE y Open Atrium tanto para gestión del proyecto,

	El curso con el que se desarrolló el caso cuasi experimental estaba compuesto por 25 estudiantes, los cuales emplearon y desarrollaron las diferentes actividades académicas relacionadas con la programación soportadas por las herramientas de CN y principios de la CA para las construcciones de los prototipos propuestos en el marco del aprendizaje de la asignatura. Dentro de la realización del proyecto se usó Diseño de la Ciencia para determinar las actividades y entregables a desarrollar, así como metodologías ágiles.	como del código generado. Es importante manifestar que también se utilizó GitHub como repositorio y gestor de retroalimentación del código generado en el proyecto.
UTPL 2	En la UTPL se desarrolló un segundo caso cuasi experimental con estudiantes de la asignatura de Programación Web con una carga horaria de 128 horas académicas correspondiente al sexto ciclo. El curso estaba compuesto por 25 estudiantes, quienes emplearon y desarrollaron las diferentes actividades académicas relacionadas con la programación de aplicaciones en contextos reales, que fueron soportadas por las herramientas de computación en la nube y principios de la CA para la construcción de los prototipos propuestos en el marco del aprendizaje de la asignatura. Así mismo se integró la metodología Mobile-D.	Spring boot, Visual Code para el desarrollo de la aplicación; así como el Tablero Kanban de Tobeio

UNL 1	El curso con el cual se desarrolló el caso cuasi experimental tres estaba compuestos por 20 estudiantes, los cuales utilizaron y desarrollaron las diferentes actividades académicas relacionadas con la programación soportadas por las herramientas de CN y principios de la CA para las construcciones de los prototipos propuestos en el marco del aprendizaje de la asignatura. El proyecto desarrollado consistió en la creación de una tienda virtual, soportada con la metodología Mobile D. Dentro del mismo caso cuasi experimental se desarrolló un prototipo para valorar la parte afectiva del comprador, el cual se ejecutaba luego de la adquisición y para medir el estado de ánimo de este.	Las herramientas de computación en la nube utilizadas en el caso cuasi experimental fueron: Project Open, Jira, StormBoard.
UNL 2	El curso con el cual se desarrolló el caso cuasi experimental tres estaba compuestos por 29 estudiantes de la asignatura Computación en la Nube del sexto ciclo con una carga académica de 120 horas.	GitHub, TeamBox, DOOPHY. Principios de CA.
UIDE 1	Programación Cliente Servidor (Programación Web y Móvil) perteneciente al Sexto ciclo de la Carrera de Informática y Multimedia; con 15 estudiantes con una carga se 120 horas académicas.	AWS, Cloud9, Diggo y GitHub. Principios de CA.
UIDE 2	Programación orientada a objetos asignatura del cuarto ciclo de 128 horas académicas, con 2 grupos de control conformado por 25 estudiantes. 1 grupo tradicional 1 grupo que uso herramientas de CN y CA.	Alfresco, Redbooth y DotProject. Principios de CA.

En el capítulo 6 se presenta a detalle los experimentos cuasiexperimentales antes mencionados.

5.4 Instrumentos utilizados

En esta etapa el objetivo fue recolectar, clasificar y medir la información obtenida mediante técnicas e instrumentos de investigación. Entre los instrumentos utilizados están:

5.4 .1 Observación

La observación es una técnica de investigación que implicó la recolección sistemática de información mediante la percepción directa de fenómenos, comportamientos o eventos. Se aplicó la observación en los 6 casos cuasiexperimentales. Los pasos desarrollados fueron:

1. Determinar el objeto, situación, caso que se va a observar.

Objetivo:

Diseñar un Modelo Pedagógico basado en herramientas de CN y CA para el desarrollo de las habilidades de programación en el área Informática.

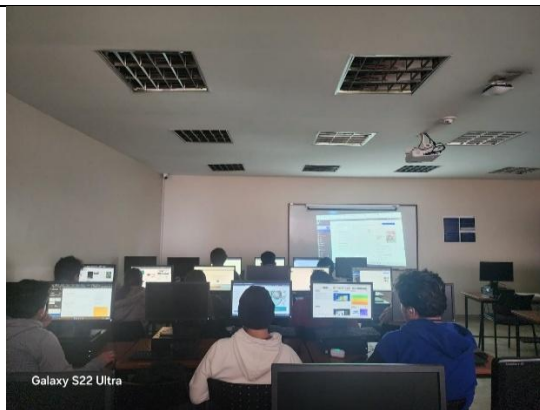
Situación:

Desarrollo de 6 casos cuasiexperimentales en 3 IES ecuatorianas de estudiantes que cursan las carreras de ciencias de la computación, informática y sistemas,

Observación:

El desarrollo de las habilidades de programación de los estudiantes participantes en los experimentos cuasiexperimentales realizados. Los datos fueron recolectados mediante herramientas de CN y fotografías de los grupos participantes. En la tabla 18 se encuentra el registro fotográfico de los participantes.

Tabla 18
Registro fotográfico de desarrollo de casos cuasiexperimentales



Observación grupo de control estudiantes UIDE 1



Observación grupo de control estudiantes UIDE 2



Observación Grupo de Control UTPL 1



Observación Grupo de Control UTPL 2



Observación Grupo de control UNL 1



Observación Grupo de control UNL 2

En la sección de anexos muestra la ficha de observación ejecutada en cada uno de los casos cuasi experimentales. Esta ficha (Anexo 1) está diseñada para ser utilizada por observadores en un entorno de enseñanza de programación, ayudando a evaluar cómo las

tecnologías emergentes pueden mejorar o afectar la experiencia educativa. Entre los resultados generados por las 139 fichas tenemos:

Análisis por Sección de la Ficha

1. Objetivos de la Clase

- **Descripción Breve:**

La clase se enfocó en la introducción al uso de plataformas de CN en entornos de desarrollo de software.

Datos clave:

- El 85% de los estudiantes logró identificar los servicios básicos ofrecidos por la plataforma seleccionada.
- Un 10% presentó dificultades en entender la integración de servicios específicos.
- Alineación: 5% totalmente alineada con el módulo herramientas de CN del programa general.

2. Herramientas de CN

- **Plataformas y Servicios Utilizados:**

Google Cloud y AWS fueron las plataformas seleccionadas.

Resultados:

- El 80% de los estudiantes interactuó correctamente con ambas plataformas.
- El 20% experimentó problemas técnicos (configuración de cuentas y acceso a los servicios).
- Accesibilidad: La principal dificultad reportada fue el tiempo de carga y latencia en algunos servicios.
- Oportunidad: Incluir un taller previo sobre configuración inicial.

3. CA

- **Herramientas y Técnicas Utilizadas:**

No se utilizaron dispositivos avanzados de análisis afectivo, pero se observó la participación emocional a través de encuestas rápidas.

Resultados emocionales:

- El 85% de los estudiantes mostró una actitud positiva durante la clase, reflejada en el interés y las preguntas formuladas.
- El 15% presentó signos de frustración ante los problemas técnicos.

4. Interacción y Participación

- Dinámicas de Clase:

La clase se organizó en grupos de cinco, fomentando el trabajo colaborativo.

Datos:

- El 90% de los grupos participó activamente en la resolución de problemas.
- Se identificaron tres grupos con bajo nivel de interacción, lo que sugiere un seguimiento más cercano del docente.
- Interacción docente-estudiantes: Hubo una media de cinco interacciones directas por grupo.

5. Evaluación y Feedback

- Métodos de Evaluación:

Se aplicó una evaluación formativa a través de un cuestionario automatizado en Google Forms.

Resultados:

- Tasa de aprobación: 88%.
- Feedback instantáneo: El 95% de los estudiantes valoró positivamente el recibir retroalimentación inmediata tras completar el cuestionario.

6. Desafíos y Oportunidades

- Desafíos:

- Dificultades técnicas al acceder a los servicios de nube (reportado por un 20% de los estudiantes).
- Limitada infraestructura tecnológica en algunos casos.

- Oportunidades:
 - Implementar una guía previa de configuración de cuentas en la nube.
 - Considerar el uso de simuladores locales en caso de problemas de conexión.

7. Comentarios Generales

- La clase fue bien recibida, con un índice de satisfacción del 80% según una encuesta de cierre.
- Se destacó el valor de trabajar con herramientas reales y el interés generado por la CN.
- Recomendación: Realizar una evaluación diagnóstica al inicio de próximas clases para adaptar mejor el ritmo.

Las plataformas de CN fueron apropiadamente utilizadas, aunque persisten desafíos técnicos que pueden mitigarse con planificación anticipada. El enfoque centrado en la participación, el feedback inmediato y la integración paulatina de la computación afectiva resultó beneficioso para el desarrollo de habilidades en programación.

5.4.2 Entrevista

La entrevista es un método de comunicación que involucra a dos o más personas, donde una parte (el entrevistador) hace preguntas y la otra parte (el entrevistado) proporciona respuestas. Es una herramienta utilizada en varios contextos, como el periodismo, la investigación y el ámbito laboral. Su objetivo principal es obtener información, opiniones o evaluar habilidades y competencias de los participantes. En el anexo 2 se detalla la entrevista aplicada en el contexto de la presente tesis doctoral.

Análisis de Entrevistas para Clases de Programación Informática

1. Experiencia General

Patrones observados:

- El 80% de los entrevistados describió su experiencia general como “positiva” debido a la accesibilidad y flexibilidad que ofrecen las herramientas en la nube.

- Un 20% reportó frustración inicial por la curva de aprendizaje al usar plataformas como AWS o Google Cloud, especialmente en la configuración de entornos y permisos.

2. Herramientas de CN

Plataformas utilizadas:

- Google Cloud (70%), AWS (50%) y Azure (30%) fueron las plataformas más mencionadas.
- Efectos en el aprendizaje:
 - El 90% de los estudiantes afirmó que la computación en la nube facilitó el acceso a entornos reales de desarrollo, mejorando su comprensión de conceptos avanzados.
 - Un 10% mencionó problemas técnicos, como caídas de conexión o restricciones en el uso de ciertos servicios gratuitos.

3. CA

Experiencia de uso:

- Solo el 20% de los entrevistados afirmó haber experimentado tecnologías de CA como análisis de expresión facial y encuestas emocionales en tiempo real.
- Impacto percibido:
 - Los que experimentaron estas tecnologías reportaron una mayor interacción en clase y una mejor empatía del docente al adaptar el ritmo de la clase según el estado emocional de los estudiantes.
 - Un profesor mencionó que el análisis afectivo le ayudó a detectar cuándo un grupo de estudiantes estaba frustrado, permitiéndole intervenir a tiempo.

4. Integración Tecnológica

Percepción de la integración:

- El 85% de los entrevistados considera que la integración de estas tecnologías en el currículo ha sido satisfactoria.
 - Motivo: La mayoría de los cursos sigue enfocándose en herramientas locales o entornos tradicionales, limitando el uso de CN y CA

- Desafíos enfrentados:
 - Conectividad inestable y falta de recursos tecnológicos (mencionado por el 30% de los entrevistados).
 - Soluciones propuestas:
 - Implementar entornos simulados cuando el acceso a internet sea limitado.
 - Aumentar la capacitación técnica del personal docente.

5. Impacto en el Aprendizaje y Participación

Participación mejorada:

- El 75% afirmó que la participación en clase aumentó con el uso de estas tecnologías, principalmente porque los entornos colaborativos en la nube permitieron un aprendizaje más dinámico.
- Motivación:
 - El 80% de los estudiantes afirmó que el uso de estas tecnologías les motivó a aprender más sobre programación avanzada, al ofrecerles una experiencia más cercana a entornos laborales reales.

7. Futuro y Mejoras

Capacitación continua: Tanto estudiantes como profesores solicitaron más talleres y cursos prácticos sobre el uso de herramientas de CN y CA.

Mejor infraestructura: Proveer mejor acceso a internet y equipos adecuados en la institución.

- Estandarización:
 - El 90% de los entrevistados visualiza un futuro donde estas tecnologías sean componentes estándar en la educación en programación, destacando que la industria requiere cada vez más competencias en entornos de nube.
 - Un pequeño porcentaje (10%) mostró escepticismo, argumentando que no todas las instituciones cuentan con los recursos para adoptar estas tecnologías.

Finalmente, en el anexo 3 se presenta la evaluación a los códigos desarrollados por los estudiantes participantes en los casos cuasiexperimentales.

Los resultados obtenidos de la evaluación de código se presentan a continuación:

1. Revisión Inicial: Objetivo del Proyecto

- Comprensión del Propósito del Proyecto: Los proyectos desarrollados abordan una variedad de problemas, desde aplicaciones de gestión de tareas hasta soluciones de comercio electrónico y plataformas de aprendizaje.
- Cumplimiento de Requisitos: El 85% de los proyectos cumplieron con los requisitos funcionales establecidos. Sin embargo, algunos proyectos mostraron carencias en funcionalidades secundarias o avanzadas.
- Plataforma:
 - Aplicaciones Web: 60% de los proyectos.
 - Aplicaciones Móviles: 30% de los proyectos.
 - Aplicaciones Híbridas: 10% de los proyectos.

2. Análisis de Estructura del Código

- Organización de Carpetas y Archivos: La mayoría de los proyectos sigue una estructura lógica, separando controladores, vistas y modelos. No obstante, el 25% presenta estructuras confusas, lo que dificulta el mantenimiento del código.
- Buenas Prácticas en Aplicaciones Móviles: El 70% de los proyectos móviles emplea patrones como MVVM y MVC correctamente.
- Nombres de Archivos y Carpetas: Se observó que el 80% de los proyectos sigue un estándar de nomenclatura consistente
- Un 20% de los proyectos presenta nombres genéricos que dificultan la comprensión del código.

3. Estilo de Código

- Legibilidad: El 90% de los proyectos presenta un código legible, con uso adecuado de sangrías y espaciado.

- Nombres Significativos: El 85% de los estudiantes utiliza nombres descriptivos para variables, funciones y clases.
- Comentarios: El 70% incluye comentarios útiles, aunque algunos proyectos presentan comentarios redundantes o innecesarios.

4. Calidad del Código

- Modularidad: El 75% del código está dividido en módulos reutilizables.
- Evitar Redundancias: Un 30% de los proyectos presenta código repetido que podría refactorizarse.
- Buenas Prácticas: El 95% utiliza Let y Const en lugar de Var en JavaScript. En aplicaciones móviles, se optimizan los ciclos de vida de actividades y Fragments en un 80% de los casos.

5. Funcionalidad

- Pruebas de Uso: El 85% de los proyectos cumple con las funciones descritas, aunque se identificaron errores menores en algunos casos.
- Flujo de Usuario: La navegación es fluida en el 90% de los proyectos.
- Adaptabilidad: El 95% de las aplicaciones web es responsive y el 85% de las aplicaciones móviles es compatible con distintas resoluciones.

6. Seguridad

- Validación de Datos: Solo el 60% de los proyectos valida correctamente el input del usuario en Frontend y Backend.
- Autenticación y Autorización: El 70% implementa correctamente la autenticación mediante sesiones o tokens, aunque solo el 40% maneja adecuadamente la expiración de tokens.

7. Optimización y Desempeño

- Eficiencia del Código: El 80% evita consultas o cálculos innecesarios.
- Carga y Velocidad: El 85% de las aplicaciones web optimiza recursos (imágenes, scripts, CSS) para mejorar la velocidad de carga.

8. Conformidad con Tecnologías Usadas

- Frameworks y Librerías: El 90% de los proyectos aprovecha correctamente las capacidades de Frameworks como React, Angular y Flutter. Un 10% incluye dependencias innecesarias que podrían eliminarse.
- Estándares del Lenguaje: El 95% sigue las convenciones modernas del lenguaje, como ES6+ en JavaScript.

9. Pruebas Automatizadas

- Cobertura de Pruebas: Solo el 40% de los proyectos incluye pruebas automatizadas.
- Calidad de las Pruebas: La calidad de las pruebas es variable y en muchos casos no se evalúan situaciones extremas o de error.

10. Retroalimentación

Logros:

- Se destaca la creatividad y el correcto uso de tecnologías modernas en la mayoría de los proyectos.
- La mayor parte del código es legible, modular y sigue buenas prácticas.

Áreas de Mejora:

1. Refactorización: Refactorizar código redundante en un 30% de los proyectos.
2. Seguridad: Mejorar la validación de datos y la gestión de sesiones/tokens.
3. Pruebas: Aumentar la cobertura de pruebas automatizadas y su calidad.

Ambas metodologías se complementan para garantizar un modelo pedagógico basado tanto en la innovación como en la evidencia.

El uso combinado de la metodología de DS y la Revisión Sistemática de la Literatura proporcionó un enfoque sólido y estructurado para desarrollar un modelo pedagógico innovador. Estas metodologías ayudan a garantizar que el modelo esté basado en evidencia científica, sea relevante en el contexto actual y responda a necesidades reales de los estudiantes. Ambas metodologías, al trabajar en sinergia, garantizan un enfoque que combina innovación, relevancia práctica y fundamentación científica.

Por un lado, la Revisión Sistemática de Literatura permitió construir una base sólida al identificar las mejores prácticas, tecnologías emergentes y brechas en el conocimiento sobre la enseñanza de programación. Esto asegura que el modelo pedagógico se alinee con las tendencias actuales, responda a las necesidades específicas de los estudiantes de ingeniería informática y tenga un respaldo empírico.

Por otro lado, el DS aporta un marco para la creación, implementación y validación de un modelo práctico, integrando herramientas de computación en la nube para accesibilidad y escalabilidad, junto con sistemas de CA que personalizan el aprendizaje en función de las emociones del estudiante. A través de ciclos iterativos, DS permite refinar continuamente el modelo, garantizando su efectividad en el desarrollo de habilidades de programación.

La combinación de estas metodologías no solo aseguró que el modelo pedagógico sea innovador y adaptativo, sino que también lo posiciona como una solución educativa integral, capaz de preparar a los estudiantes para afrontar los desafíos del entorno tecnológico actual.

Capítulo 6: Resultados

En el presente capítulo se aborda el desarrollo de los escenarios cuasi experimentales basados en la CN y CA, en el proceso de desarrollo de las destrezas de programación en estudiantes de ciencias de la computación e ingeniería informática pertenecientes a tres IES ecuatorianas que formaron parte de la presente investigación. Para el desarrollo de la fase empírica se toma como base las conceptualizaciones teóricas, programas académicos y teorías de enseñanza de la programación en contextos universitarios.

La fase cuasi experimental empírica se sustenta en las conceptualizaciones teóricas existentes, los programas académicos vigentes y las teorías de enseñanza de la programación aplicadas en contextos universitarios. Este enfoque permite analizar y diseñar estrategias didácticas que potencien las competencias de programación, integrando conocimientos y prácticas relevantes para los estudiantes.

A continuación, se detallan los casos cuasi experimentales desarrollados en la presente investigación.

6.1 Desarrollo de los escenarios cuasi experimentales

De acuerdo Giró-Gracia y Sancho-Gil (2022), el uso de la tecnología digital está impregnando y transformando todos los sistemas sociales, y la educación no es una excepción. El desarrollo de las herramientas de CN y de la CA ha dado un nuevo impulso a la esperanza de dotar a los sistemas educativos de soluciones eficaces y más personalizadas para la enseñanza y el aprendizaje. Es importante considerar que el estudiante de las áreas de ciencias de la computación necesita desarrollar su pensamiento computacional que implica la capacidad creativa, crítica y estratégica para utilizar los fundamentos de la computación en diferentes áreas del conocimiento y resolver problemas, de forma individual o colaborativa apoyado de herramientas de CN y CA.

El pensamiento computacional fue considerado dentro del desarrollo cuasi experimental, ya que permitió utilizar técnicas y herramientas de la informática, en este caso

herramientas de CN, para resolver problemas y comprender fenómenos en diferentes ámbitos. Incluye habilidades como la resolución de problemas, la programación, análisis de datos y la visualización. El pensamiento computacional también se utilizó para desarrollar habilidades críticas y creativas que son valiosas en un mundo cada vez más tecnológico.

Para Viana y Castillo (2002), el docente necesita descubrir que los estudiantes pueden aprender de diferentes maneras, por lo que es necesario diversificar metodologías, para que las clases sean más dinámicas y atractivas y, en consecuencia, los estudiantes desarrollen más su conocimiento, de ello la necesidad de integrar las herramientas de CN y la CA como parte de un modelo pedagógico de enseñanza. En este contexto, un modelo pedagógico que se base en la CN y CA se ha propuesto como un nuevo enfoque de enseñanza, utilizando diferentes métodos de Ciencias de la Computación, capaces de generar nuevos enfoques educativos, en lo que respecta a la innovación en las IES que ofertan las carreras relacionadas a las ciencias de la computación, junto con el desarrollo de competencias, en la búsqueda de soluciones de problemas que necesitan ser entendidos por una nueva generación de estudiantes inmersa en tecnologías.

Para el desarrollo cuasi experimental de la presente tesis doctoral se desarrollaron 6 casos de estudio del uso de herramientas de CN y CA con estudiantes universitarios que cursaron componentes educativos de programación en instituciones de educación superior (IES) de Ecuador, en la provincia de Loja, ciudad de Loja.

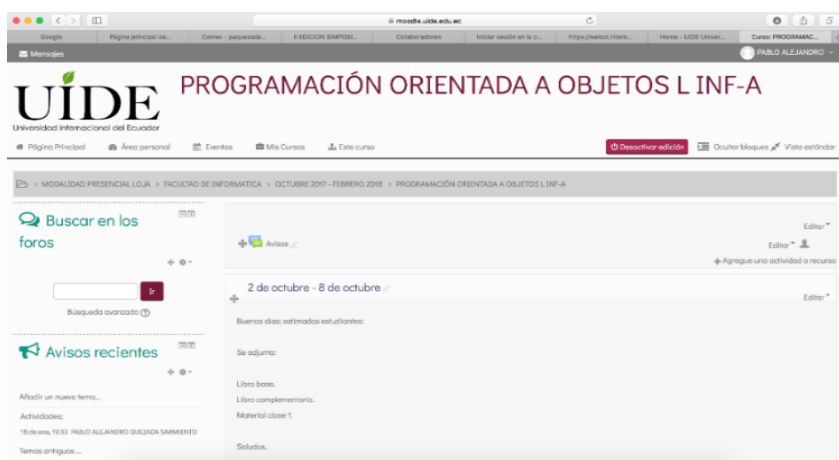
6.1.1 Caso Cuasi experimental 1: Programación Orientada a Objetos (POO) sin uso de CN y CA (Universidad Internacional del Ecuador -UIDE)

La enseñanza de POO es un enfoque pedagógico que se utiliza para dar a conocer los conceptos y principios fundamentales de la POO a los estudiantes. Aprender POO requiere tiempo y práctica. Es importante dar a los estudiantes suficiente tiempo para intercambiar ideas y practicar la implementación de POO en sus proyectos. La POO permitió a los estudiantes comprender de manera más profunda cómo los conceptos de clases, objetos, herencia, polimorfismo y encapsulamiento se aplican en la construcción de software.

Además, facilita la adquisición de habilidades necesarias para enfrentar desafíos reales en el ámbito de la programación y el desarrollo de software. Como parte del desarrollo de los casos cuasi experimentales, para comprobar el modelo propuesto, se tomó como grupo de control a 25 estudiantes del componente POO, paralelo A de la UIDE, al grupo se le enseñó la POO utilizando el estilo tradicional; con el segundo grupo de control, se aplicó el uso de las herramientas de CN y CA (Paralelo B). Además, como parte del acompañamiento se usó el aula virtual (Moodle) del componente, esto se puede apreciar en la Figura 32.

Es importante mencionar que Moodle es un sistema de gestión de cursos de código abierto (CMS), también conocido como sistema de gestión del aprendizaje (LMS) o entorno de aprendizaje virtual (VLE). Según Dougiamas y Taylor (2003), Moodle fue creado en 2002 y se hizo muy popular entre los educadores de todo el mundo para la creación en línea de sitios web dinámicos para sus estudiantes. Para trabajar, debe instalarse en un servidor web en algún lugar, ya sea en uno de sus propios equipos o en una empresa de alojamiento web. Así mismo Moodle es un espacio de aprendizaje virtual que facilita la comunicación e interacción entre estudiantes y docentes, además esta plataforma permite compartir recursos educativos y realizar actividades académicas en línea (foro, chat y video colaboración) (Iglesias et al.,2014).

Figura 32
Moodle POO



Nota. Moodle POO usado en el contexto del caso cuasi experimental

La utilización de un entorno Moodle puede amplificarse mediante la adopción de un emergente TIC. En la Figura 33, se presenta al grupo de control que fue sujeto del caso 1.

Figura 33
Grupo de Control



Nota. Grupo de control

El tiempo de desarrollo del caso de estudio fue de 6 meses, durante 2 parciales académicos, en los cuales los estudiantes desarrollaron actividades para adquirir las habilidades de programación. A los participantes se los evaluó de forma tradicional con trabajos en la plataforma y casos de estudio en los lenguajes de programación seleccionados en función del sílabo de la asignatura.

Aprender a programar puede ser difícil para los estudiantes, especialmente al principio. Estos son algunos problemas comunes que se detectaron en el desarrollo del caso cuasi empírico:

1. Comprensión de conceptos abstractos. La comprensión de conceptos abstractos implicó la habilidad de analizar, interpretar y relacionar ideas o abstracciones sin depender únicamente de la información literal o concreta.
2. Falta de práctica. La POO requiere práctica constante, leer libros, artículos o ver tutoriales no es suficiente. La mejor manera de aprender a programar es escribir códigos regularmente. Intentar resolver problemas de programación, participar en proyectos de código abierto o crear proyectos personales propios para ganar experiencia práctica.

3. Sobrecarga de información. Con la cantidad de lenguajes de programación, herramientas y tecnologías disponibles, puede ser abrumador decidir qué aprender. Se seleccionó un lenguaje de programación popular para comenzar desde allí.
4. Falta de paciencia y perseverancia. La POO puede ser desafiante y es común enfrentar errores y obstáculos en el proceso de aprendizaje. Es importante tener paciencia y perseverar a través de estos desafíos. Los errores son oportunidades de aprendizaje y ayudarán a los estudiantes a mejorar sus habilidades de programación.
5. Falta de recursos adecuados. Es importante que los estudiantes tengan acceso a buenos recursos de aprendizaje, como tutoriales en línea, libros y cursos. Además, se recomienda buscar comunidades en línea donde pueda hacer preguntas y obtener apoyo de programadores más experimentados. Es importante que los estudiantes aprovechen las plataformas de aprendizaje en línea que ofrecen cursos estructurados y proyectos prácticos.
6. Falta de planificación. Tener un plan de aprendizaje estructurado ayudó a progresar de manera más eficiente. Es importante definir metas claras y establecer un cronograma realista para alcanzarlas; esto se pudo establecer mediante el syllabus de la asignatura.

6.1.1.2 Caso cuasi experimental 1.2: POO con uso de CN Y CA (Universidad Internacional del Ecuador)

La CN es un campo en constante cambio, por lo que es importante que los estudiantes se mantengan actualizados con los últimos desarrollos y tecnologías en el campo. Los docentes deben fomentar la curiosidad y el aprendizaje continuo para que puedan adaptarse y aprovechar al máximo la CN. La enseñanza de programación con CN es una excelente opción para los estudiantes interesados en adquirir habilidades en programación y aprovechar los beneficios de la infraestructura en la nube. En este contexto el grupo de control

seleccionado para el presente caso cuasi experimental estaba integrado por 25 estudiantes de Informática y Multimedia de la UIDE que cursaron el componente educativo de POO del cuarto ciclo de la titulación antes mencionada en el periodo octubre 2017 – febrero 2018 Paralelo B (Figura 34).

Figura 34
Grupo de Control 2

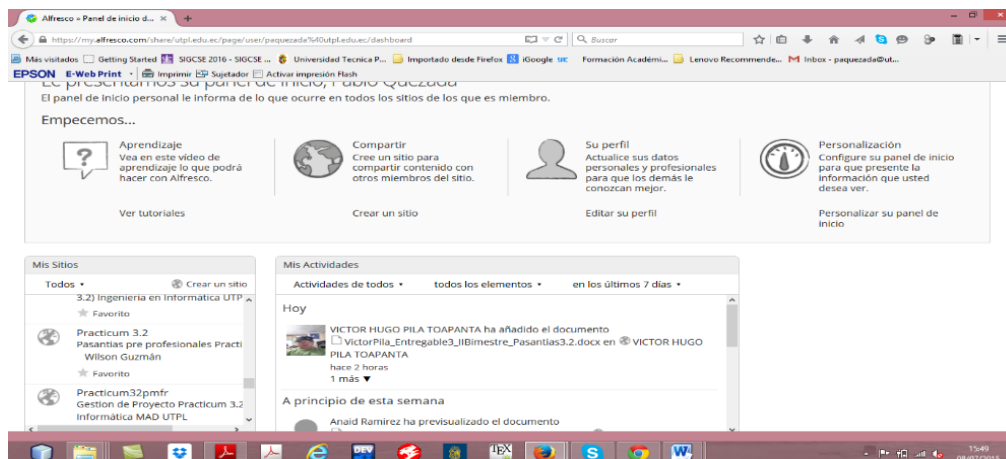


Nota. Grupo de Control Estudiantes de POO que utilizaron herramientas de CN

El tiempo de desarrollo del caso de estudio fue de 6 meses durante 2 parciales académicos. En el proceso de análisis se utilizaron las herramientas de CN: Alfresco, Redbooth, DotProject. A continuación, se detalla las características y resultados del uso de las herramientas de CN utilizadas.

Alfresco. De acuerdo con Kurysheva (2014) es una plataforma de contenido empresarial que se puede emplear en la nube o detrás de un firewall. Ayuda a almacenar y compartir diferentes tipos de documentos y recursos en la gestión empresarial. En el ámbito educativo tiene una gran aplicabilidad en la creación de repositorios y control de proyectos de aplicación que simulen un entorno real (Quezada et al., 2018) (Ver Figura 35).

Figura 35
Sitio Alfresco POO



Nota. Sitio Alfresco POO utilizado como herramientas de CN en el proceso de desarrollo de habilidades de programación.

Durante el desarrollo del uso de Alfresco en el contexto del caso experimental se establecieron las siguientes ventajas presentadas en la tabla 19.

Tabla 19
Ventajas de Alfresco

Fácil de usar.

Incremento de la productividad.

Bajo coste.

Código abierto.

Gestión avanzada de búsquedas de documentación.

Incluye sistemas de seguridad en la gestión de documentos.

Arquitectura distribuida.

Alta disponibilidad.

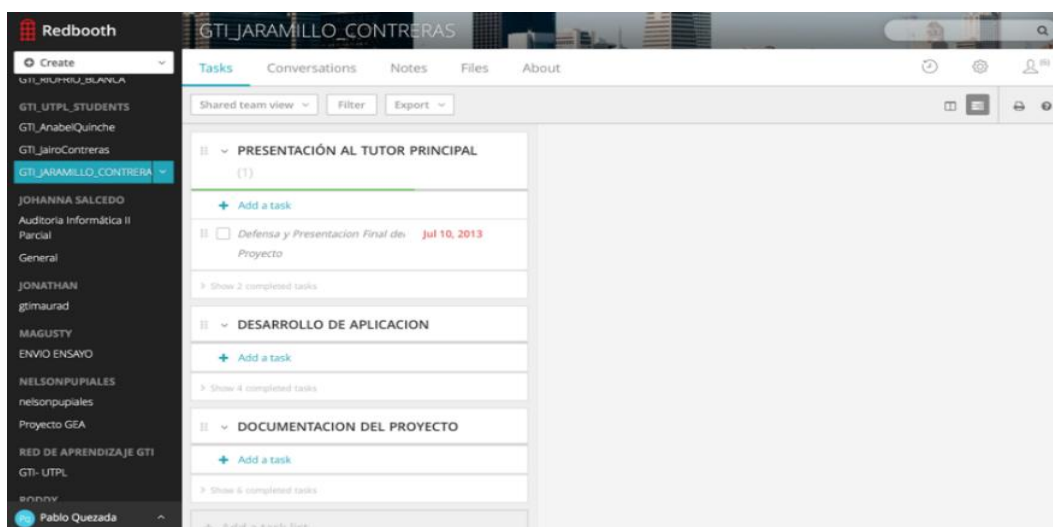
Estructura de soporte muy flexible: certificaciones, fórums, portal, etc.

Manejo de firma electrónica (Sousa et al., 2016).

Nota. Ventajas del uso de Alfresco como herramientas de CN en el contexto del desarrollo de las habilidades de programación.

Redbooth. Es otra de las herramientas de la nube utilizadas, también conocida como TeamBOX, es una plataforma de colaboración y comunicación que proporciona un lugar para las tareas compartidas, debates, intercambio de archivos, grupo de chat y videoconferencias. Es flexible y fácil de emplear, permitiendo a los equipos de proyectos y docentes realizar su trabajo de forma profesional e interactiva (Torres, et al. 2021). En el contexto de nuestro caso cuasi experimental se lo utilizó como un software de gestión de proyectos y colaboración en equipo que permitió a los usuarios gestionar tareas y proyectos con miembros del grupo en una sola plataforma. Incluye funciones como asignaciones de tareas, calendarios, uso compartido de archivos y mensajería en tiempo real. También tiene integraciones con otras herramientas de productividad como Google Drive y Dropbox. En la Figura 36, se puede apreciar la interfaz de Redbooth empleado en el presente caso cuasi empírico.

Figura 36
Red Social Generada en Redbooth



Nota. Herramientas de CN Redbooth utilizado en el caso empírico

De acuerdo con Tan & Kim (2011) Redbooth tiene una nueva versión de la plataforma de colaboración; además de los cambios evidentes a la interfaz de usuario, hay muchas otras mejoras que hacen que la plataforma sea más potente.

Dentro del caso empírico, Redbooth permitió gestionar las tareas de los proyectos de manera fácil, agrupándolas bajo listas, que tienen fecha de inicio y fin. Las tareas creadas

también disponen de una fecha tope de realización, posibilidad de cambiar entre estados (en espera, activa, completa, rechazada, etc.) y asignar la tarea a un estudiante dentro del proyecto que se encontraba en desarrollo. De esta manera, las diferentes tareas a realizar durante el desarrollo de un proyecto quedan organizadas en líneas de acción genéricas, creando nuevas o cerrando según demanda. Es importante mencionar que Redbooth se utilizó principalmente en las pequeñas y medianas empresas para mejorar la productividad del equipo y optimizar los flujos de trabajo, por lo cual cada estudiante va desarrollando sus habilidades blandas y de gestión de proyectos. El punto fuerte de Redbooth dentro del caso empírico fue la comunicación entre los participantes, ya que sobre las tareas se abre automáticamente un hilo de comentarios para que cualquier usuario adherido al proyecto pueda aportar datos o añadir réplicas (SocialMedia 2014).

DotProject. Refuerza su posicionamiento como herramienta colaborativa al permitir la generación de foros. Esto resulta útil si hay que consensuar determinadas opiniones, considerando que se abrirían distintos foros de acuerdo con las necesidades de los miembros (EcuRed, 2020).

En la tabla 20 se presentan las características de las herramientas utilizadas en el caso experimental. Los criterios que se consideraron para seleccionar las herramientas de CN fueron analizados considerando las características expuestas en la tabla antes mencionada.

Tabla 20
Características de las Herramientas Cloud utilizadas

Características	Alfresco	Redbooth	DotProject
Compartir, gestionar tareas.	Sí	Sí	Sí
Componente colaborativo.	Sí	Sí	Sí
Integración con herramientas en la nube.	Sí	Sí	Sí
Notificaciones vía mail.	Sí	Sí	Sí
Invitaciones y generación de redes colaborativas.	Sí	Sí	Sí
Gestión de usuarios.	Sí	Sí	Sí
Accesibilidad.			
Licencia de Pago.	Sí	Sí	No

Nota. Características de las Herramientas Cloud utilizadas adaptada de Quezada et. al (2015)

Las características detalladas en la tabla 20 permiten establecer que las herramientas de CN ofrecen prestaciones en el contexto educativo y de gestión de proyectos basados en principios de innovación e ingeniería de software (Quezada et al., 2015).

En la tabla 21 se muestra la frecuencia de uso de las herramientas (horas promedio entre el número de participantes) por parte del público de la investigación desarrollada en siete proyectos de software (considerando la naturaleza del proyecto).

Tabla 21
Características de las Herramientas Cloud utilizadas en los proyectos de software

Proyecto de Software	Alfresco	Redbooth	DotProject
Calculadora Virtual	3	1	2
Security	1	1	1
Saraguro Tourist	2	1	2

Gestor académico basado en aspectos	1	2	1
Web Semántica	1	1	1
Recorrido de rutas virtuales			
Célica 2.0	2	2	2
Gestor Vehicular basado en aspectos. "Garage 2.0"			
Total	10	8	10

Nota. Frecuencia de Uso de Herramientas de Computación en la Nube

En la tabla 22 se detalla la interactividad y Feedback generado entre docente y estudiantes sujetos del presente estudio.

Tabla 22
Interactividad y Retroalimentación generada

Proyecto de Software	Número de participaciones	Alfresco	Redbooth	DotProject
Calculadora Virtual	16	15	16	16
Security	14	11	10	14
Saraguro Tourist	14	13	11	14
Gestor académico basado en aspectos	12	11	9	12
Web Semántica	14	13	8	14
Recorrido de rutas virtuales	11	10	9	11
Célica 2.0.				
Gestión Vehicular basada en aspectos. "Garage 2.0".	19	22	17	20
Total	100	95	80	101

Nota. Datos de la interactividad en las herramientas de CN

A continuación, se detallan los resultados obtenidos en el desarrollo del caso experimental. En la tabla 23 se presentan los resultados generados del uso de las herramientas de CN utilizadas en caso cuasiexperimental.

Tabla 23

Resultados del uso de las herramientas de CN

Mejora en la gestión por la frecuencia de empleo en los de proyectos de software.

Mayor interactividad entre docente y profesionales en formación.

Gestión para la toma decisiones.

Acceso a la información de cada uno de los proyectos desarrollados.

Vinculación de cuentas, accesos y notificaciones de acuerdo con cada herramienta.

Generación de portafolios digitales y Feedback del componente educativo.

Nota. Resultados obtenidos en el caso cuasiexperimental.

6.1.2 Caso cuasiexperimental.2: Uso de herramientas de CN GitHub en la Universidad Nacional de Loja.

En el contexto del desarrollo del caso cuasi experimental 2 se trabajó con estudiantes de la Universidad Nacional de Loja (UNL) que cursaron la asignatura de Computación en la Nube, ubicada en el sexto ciclo de la carrera de Computación, con una carga de 120 horas académicas. El curso con el cual se desarrolló el caso experimental estaba compuesto por 20 estudiantes que utilizaron y desarrollaron diferentes actividades académicas relacionadas con la programación, soportadas por las herramientas de CN y principios de la CA para la construcción de los prototipos propuestos en el marco del aprendizaje de la asignatura.

En la Figura 37, se presenta el grupo de control que desarrolló el caso cuasi experimental 2.

Figura 37
Grupo de Control



Nota. Estudiantes del grupo de control caso cuasi experimental 2

6.1.2.1 Descripción del repositorio utilizado

GitHub es un Open Source Software (OSS) fundada en 2008 con el propósito de simplificar el intercambio de código (XACATA,2021). Los OSS hoy en día son un tipo de software extremadamente popular entre la comunidad de programadores y usuarios de computadoras y parece que esta tendencia se intensificará en los próximos años. En un alto nivel, GitHub es un sitio web y un servicio basado en la nube que ayuda a los desarrolladores a almacenar y administrar su código, así como a rastrear y controlar los cambios en su código. La herramienta GitHub es una plataforma de desarrollo inspirada en el trabajo colaborativo. Los elementos esenciales de GitHub son presentados en la tabla 24.

Tabla 24
Elementos GitHub

Repositorios
Ramas
Se compromete
Solicitudes de extracción
Git (el software de control de versiones en el que se basa GitHub)

Se puede usar un repositorio de GitHub para almacenar un proyecto de desarrollo (XACATA,2021).

Puede contener carpetas y cualquier tipo de archivo (HTML, CSS, JavaScript, Documentos, Datos, Imágenes).

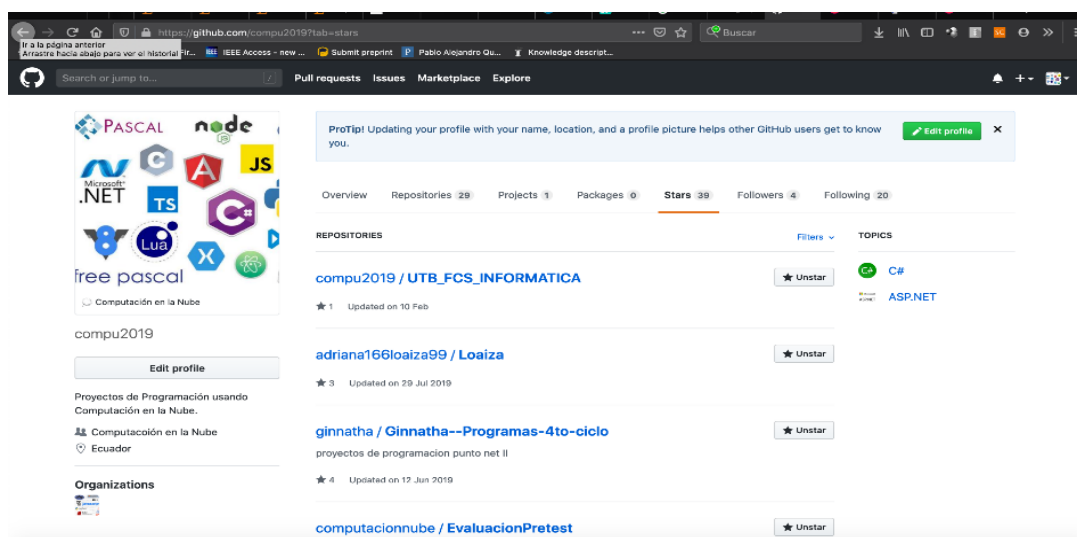
Nota. Elementos GitHub

Un repositorio de GitHub también debe incluir un archivo de licencia y un archivo README sobre el proyecto. GitHub también se puede utilizar para almacenar ideas o cualquier recurso que se desee compartir (w3schools.com, 2020).

En la Figura 38 se puede visualizar el repositorio utilizado en el presente caso de cuasiexperimental

Figura 38

Repositorio GutHub Computación en la Nube en Marco del Proyecto de la tesis doctoral



Nota. Repositorio GutHub

De acuerdo con Tarigan et al. (2019), la gestión del código fuente es una de las herramientas más vitales en el desarrollo de software moderno. GitHub es uno de los sistemas de control de versiones basados en web más populares que ofrece control de versiones y administración de código fuente.

Entre las características más sobresalientes de esta herramienta están:

Rama. Una rama de GitHub se usó para trabajar con diferentes versiones de un repositorio al mismo tiempo. Por defecto, un repositorio tiene una rama maestra (una rama

de producción). Cualquier otra rama es una copia de la rama maestra (como lo fue en un punto en el tiempo) (Dabbish et al.,2012).

Commits. En GitHub los cambios se llaman Commits. Cada confirmación (cambio) tiene una descripción que explica por qué se realizó. De acuerdo con Ko et al. (2007).

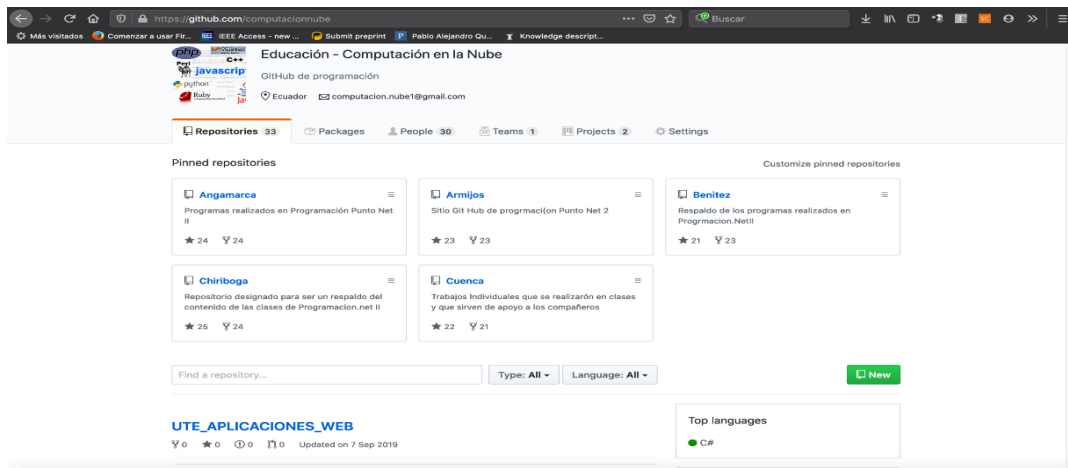
Solicitudes de extracción. Las solicitudes de extracción son el corazón de la colaboración de GitHub. Con una solicitud de extracción se está proponiendo que sus cambios se fusionen con el maestro. Las solicitudes de extracción muestran diferencias de contenido, cambios, adiciones y sustracciones en colores (verde y rojo) (Cammariere et al.,2016).

Control de versiones Git. El control de versiones ayuda a los desarrolladores a rastrear y administrar los cambios en el código de un proyecto de software. A medida que crece un proyecto de software, el control de versiones se vuelve esencial. En cambio, el control de versiones permite a los desarrolladores trabajar de manera segura a través de la ramificación y la fusión.

Con la ramificación, un desarrollador duplica parte del código fuente (llamado repositorio). El desarrollador puede realizar cambios de forma segura en esa parte del código sin afectar el resto del proyecto. Una vez que el desarrollador hace que su parte del código funcione correctamente puede fusionarlo nuevamente en el código fuente principal para hacerlo oficial (Cicciarella et al.,2015).

En la Figura 39 se puede visualizar el GitHub de CN del componente educativo de programación; el cual fue usado por uno de los grupos de control como herramienta complementaria en el proceso de formación profesional.

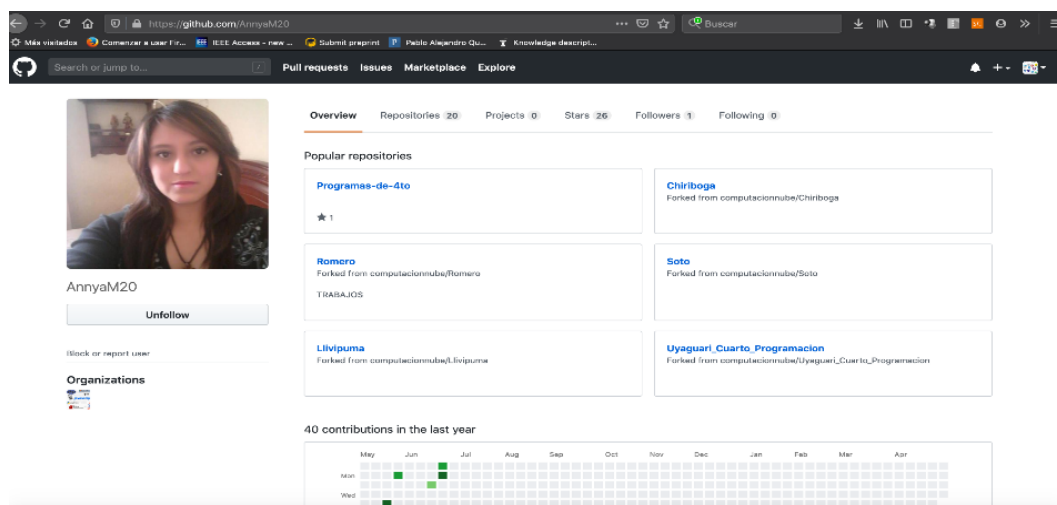
Figura 39
Herramientas de CN GitHub



Nota. GitHub del Componente Educativo

Grupo de control. 20 estudiantes que cursaron el componente educativo de programación, de la carrera de Informática del p nsum de estudios de Ingenier a, y que como grupo de control utilizaron el Git Hub como herramienta en su proceso de formaci n, para la adquisici n de las habilidades de programaci n. Cada estudiante cre  su propio repositorio donde subi  los trabajos y proyectos respectivos. En la Figura 40, se muestra uno de los repositorios implementados por los estudiantes que formaron parte del caso experimental.

Figura 40
Repositorio Implementado

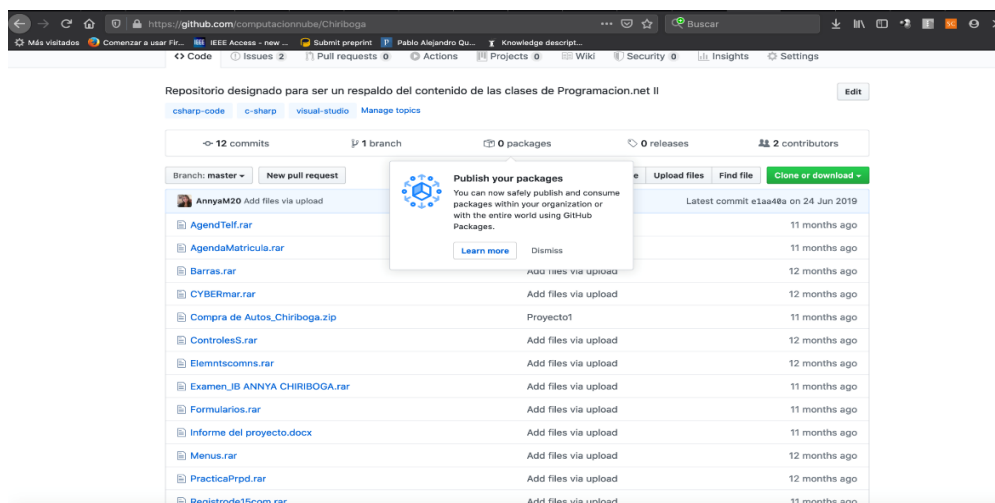


Nota. Ejemplo de GitHub desarrollado por los participantes del caso cuasiexperimental

La comunidad de desarrollo de software del caso experimental ha adoptado a GitHub como una plataforma esencial para administrar sus proyectos de software. Durante el uso de GitHub se crearon eficiencias que mejoraron la forma de trabajo de los profesionales de software. No solo proporciona un repositorio de proyectos rastreable, sino que actúa como un lugar de encuentro social para las partes interesadas, apoyando a las comunidades de práctica. Zagalsky et al. (2015) manifiesta que los educadores han visto el potencial de las funciones de colaboración de GitHub para administrar y mejorar, tal vez incluso transformar, la experiencia de aprendizaje.

En la Figura 41 se muestra parte de la actividad desarrollada por el estudiante. Cada estudiante subió las diferentes actividades realizadas en la asignatura de programación.

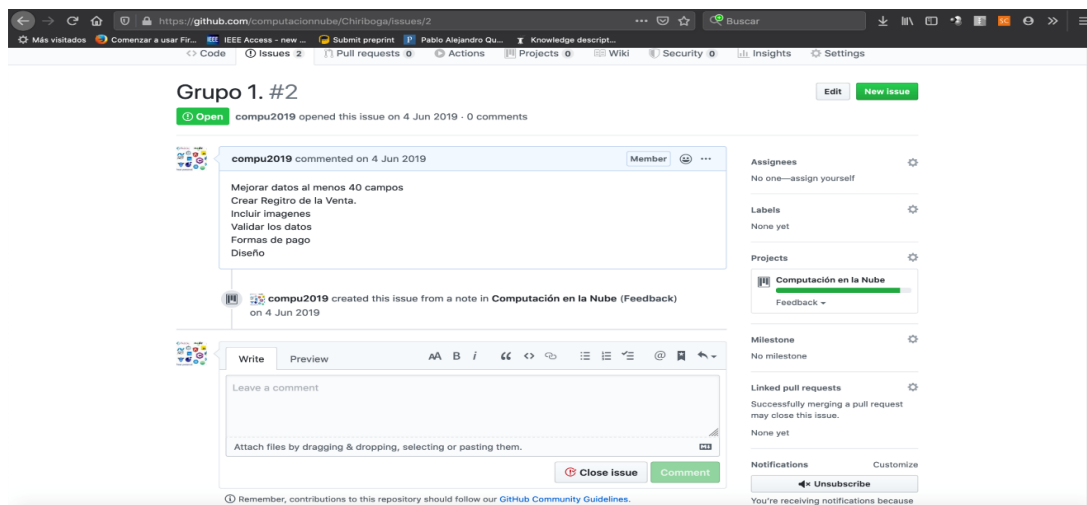
Figura 41
Repositorio de programas desarrollados



Nota. Repositorio de programas desarrollados.

En el mismo contexto en la Figura 42 se puede revisar la retroalimentación respecto al trabajo colaborativo que se generó en uno de los proyectos.

Figura 42
Retroalimentación de proyectos



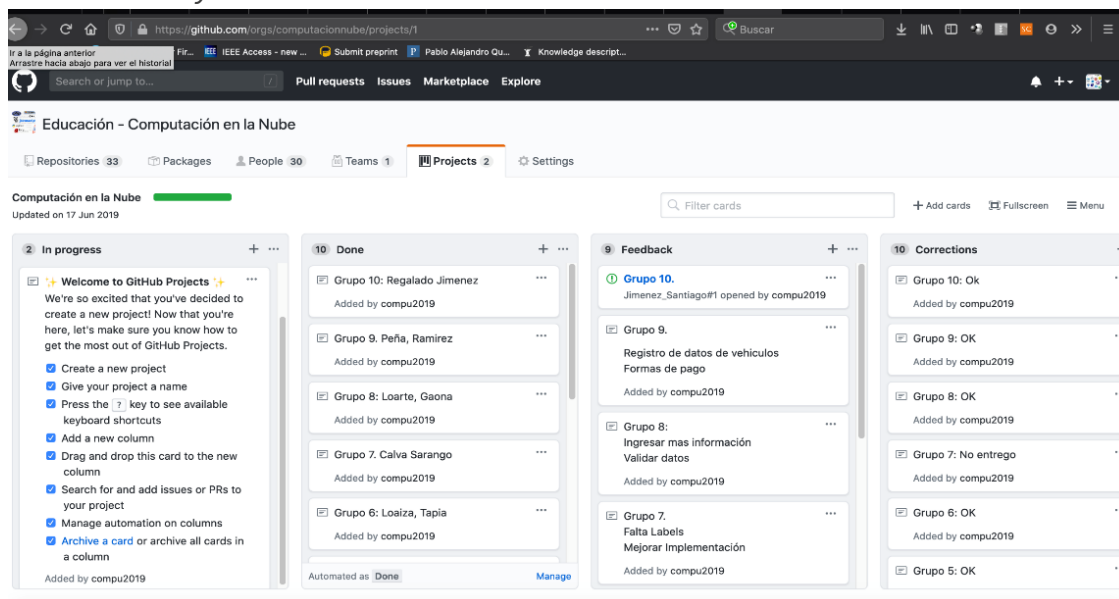
Nota. Retroalimentación Git centro de operaciones

Es importante destacar que como parte del desarrollo del componente educativo se generaron espacios de discusión en línea.

Dependiendo de cómo esté configurado el repositorio, también puede crear su propia rama y, una vez que haya realizado algunos cambios, puede enviar ese código a una sucursal mediante una solicitud de extracción.

En la Figura 43 se puede apreciar el control de los proyectos desarrollados en el caso cuasi experimental 3.

Figura 43
Control de Proyectos



Nota. Control de proyectos de Software mediante GitHub

Durante el desarrollo del caso cuasi experimental se determinó que los estudiantes realizaron un conjunto de inferencias sociales a partir de la información de actividad en red en GitHub, como inferir los objetivos técnicos y la visión de otro estudiante cuando editan código o adivinar cuál de varios proyectos similares tiene la mejor oportunidad de prosperar a largo plazo. Los estudiantes combinaron estas inferencias en estrategias efectivas para coordinar el trabajo, mejorar las habilidades técnicas y administrar su reputación.

Así, mismo GitHub les permitió alojar y colaborar en proyectos de software utilizando el sistema de control de versiones Git. Además, GitHub es una excelente forma de desarrollar habilidades de colaboración y trabajo en equipo en un entorno en línea

Dentro del desarrollo del proyecto también se utilizaron las siguientes herramientas de CN:

TeamBOX permitió administrar proyectos, como un Facebook para la oficina; es una solución para mejorar la productividad, su principal característica es que permite crear proyectos con diferentes personas, verlos y compartir información directamente con los participantes, subiendo archivos y viendo el tiempo empleado para cada tarea.

La siguiente herramienta de CN utilizada fue Bitrix24, un gestor en línea que ayudó a planificar, comprobar y controlar los proyectos, y tareas, de la presente práctica de innovación docente. Esta herramienta permitió definir los objetivos, colaborar, trabajar en equipo y seguir el progreso de los proyectos.

Los factores que se consideraron para seleccionar estas herramientas de CN fueron:

Mejorar la coordinación y la productividad de los equipos de trabajo, lo cual permitió distribuir el trabajo entre los profesionales en formación del componente educativo y registrar el esfuerzo realizado por cada uno generando un ambiente interactivo en un entorno virtual.

Las herramientas de CN utilizadas ayudaron con la comunicación, la colaboración y el trabajo en equipo ya que se pudo compartir cualquier tipo de archivo o información con el resto de los usuarios.

6.1.3 Caso cuasi experimental 3. Prototipos Web Desarrollados mediante herramientas de CN (Universidad Técnica Particular de Loja)

En el contexto del desarrollo del caso cuasi experimental 3 se trabajó con estudiantes de la Universidad Técnica Particular de Loja que cursaron la asignatura de Programación Avanzada, del cuarto ciclo de la carrera de Computación con una carga horaria de 4 créditos; considerando que cada crédito equivale a 32 horas académica. El curso con el cual se desarrolló el caso experimental estaba compuesto por 25 estudiantes, que utilizaron y desarrollaron las diferentes actividades académicas relacionadas con la programación soportadas por las herramientas de computación en la nube y principios de la CA para la construcción de los prototipos propuestos en el marco del aprendizaje de la asignatura.

En la Figura 44, se puede apreciar al grupo que participó en el caso cuasiexperimental.

Figura 44
Grupo de Control cuasiexperimental



Nota. Grupo de control Caso Cuasi experimental 3

Es importante considerar que las TIC actualmente son herramientas para apoyar las actividades de las organizaciones y permitir una respuesta óptima y rápida a los procesos de los sistemas implantados. Todos los sectores económicos se ven actualmente afectados en mayor o menor medida por las TIC, y el sector turístico no está excluido de los cambios de las nuevas tecnologías, por lo que se pretende potenciar el turismo mediante el desarrollo de un portal web, redes sociales y marketing digital del cantón Yacuambi.

Para el desarrollo de este caso cuasi experimental, se utilizó como metodología de investigación Diseño de la Ciencia, ya que nos proporcionará un proceso de investigación constructivista, organizado, bien definido y documentado, apoyado por los procesos y resultados esperados (artefactos). El proceso de DC incluyó sus seis fases: identificación y motivación del problema, definición de los objetivos de una solución, diseño y desarrollo, demostración, evaluación y comunicación.

Dentro del proceso de investigación, basado en el DC, se desarrollaron las actividades propuestas por esta metodología que se describen en la tabla 25.

Tabla 25
Actividades propuestas en la Metodología de DC

Actividad 1. Identificación de problemas y motivación: En esta actividad definimos el problema específico de la investigación y justificamos el valor de una solución en el contexto educativo utilizado en paradigmas computacionales.

Actividad 2. Definir los objetivos de una solución. Se plantearon los objetivos de esta investigación, que fueron la base para generar resultados y artefactos basados en el diseño científico.

Actividad 3. Diseño y desarrollo: En esta actividad el artefacto fue desarrollado, en este caso el sitio web de la ciudad de Yacuambi.

Actividad 4. Demostración: El artefacto a desarrollar (página web) trata de resolver el problema de la promoción del turismo en la ciudad de Yacuambi.

Actividad 5. Evaluación: el artefacto admite una solución al problema, comparamos los objetivos de la solución y los resultados reales observados del uso de la demostración de artefacto.

Actividad 6. Comunicación: como resultado del proceso de investigación se obtendrá una publicación científica.

El prototipo desarrollado se basó en el uso de herramientas de CN y de colaboración en línea.

Nota. Actividades propuestas en el Diseño de la Ciencia en el caso cuasi experimental

6.1.3.1 Análisis y Planificación del Sitio Web desarrollado mediante el soporte de herramientas de CN.

Lo primero que realizaron los participantes cuando iniciaron el trabajo en el proyecto fue un análisis básico de la página. Este análisis nos permitió saber lo que hacen los visitantes en una página web y con estos datos se pudo tomar la mejor decisión para optimizar el contenido y así elegir un software ágil para el desarrollo del proyecto; ya que en se necesitaba editores para diseños web, servidores web, sistemas operativos y otros mecanismos fundamentales para el desarrollo.

Luego se presentó, paso a paso, cómo se planificó y organizó el desarrollo de una página, detallando cada aspecto planteado para el proyecto, partiendo de una entrevista a los propietarios y finalmente con el proyecto terminado.

Para desarrollar la web, se propuso una entrevista con los propietarios para informarles y detallar el objetivo del proyecto, con el fin de transmitir un mensaje a la comunidad y promover el turismo de Zamora.

Los argumentos analizados en la entrevista hicieron posible tener una idea reforzada de los siguientes elementos:

- Objetivo. Determinar los objetivos del proyecto, detallando sistemáticamente lo que se quiere lograr.
- Audiencia. Este elemento es muy importante, ya que la aceptación de una página web depende de ellos.
- Tecnología. Elegir un software para el diseño y desarrollo de la página.
- Encuesta de requisitos de usuario.

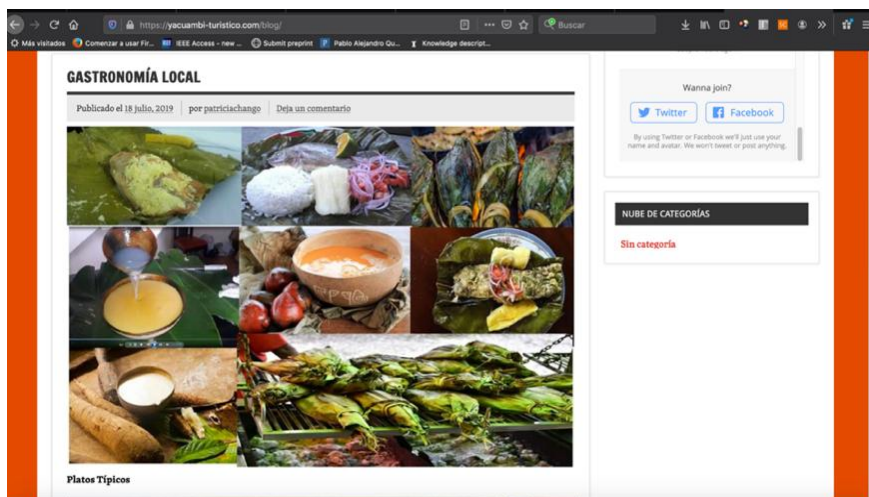
Todos estos parámetros se presentaron en el primer boceto de diseño que fue analizado por los propietarios y corregido en las siguientes visitas.

Con la información lista y con la estructura de la página web, se establecieron las propiedades y validaciones comunes que cuentan con los formularios web, menús de

opciones e informes que permitieron la interacción entre el usuario y la página web (Quezada et al.,2020).

La Figura 45 muestra el sitio implementado en el marco del proyecto propuesto.

Figura 45
Sitio Web desarrollado



Nota. Sitio Web Desarrollado tomado de Quezada et al., (2020)

Como herramientas de CN se utilizó EGROUWARE, que es una solución de trabajo en grupo vía web, de código abierto. Está escrita en PHP utilizando bases de datos, tales como LDAP, PostgreSQL o MySQL.

Así mismo se empleó REDMINE, una herramienta para la gestión de proyectos y el seguimiento de errores, escrita usando el framework Ruby on Rails. Como base de datos soporta tanto MySQL como PostgreSQL o SQLite.

Como tercera herramienta se utilizó Open Atrium, una herramienta que incluye, entre otras utilidades una wiki, un gestor documental y un gestor de incidencias, en este caso de los bugs generados en el proyecto. Open Atrium es una herramienta open source, altamente personalizable construida sobre Drupal y WordPress, con una comunidad in crescendo y que, aunque su verdadero potencial se muestra con su utilización dentro de grupos bastante poblados, se adapta también a equipos pequeños.

6.1.4 Caso Cuasi experimental 4 Desarrollo de aplicativo Web y Móvil (Universidad Técnica Particular de Loja)

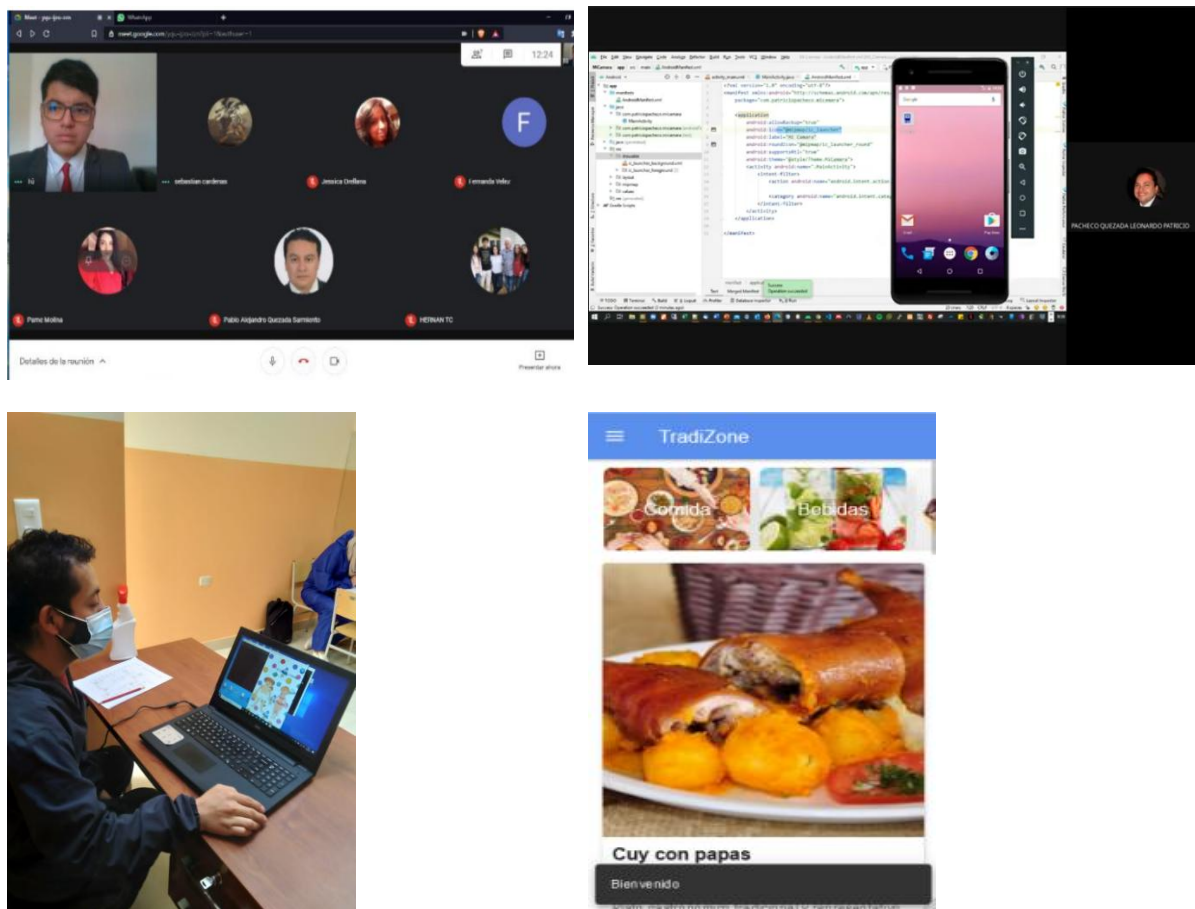
El uso de las TIC en la educación se está enfocando en la enseñanza, promoviendo un aprendizaje significativo a través de herramientas tecnológicas como las CA, especialmente en el nivel de educación superior donde el estudiante dispone de diferentes herramientas de carácter colaborativo, basadas en tecnologías 3.0 que facilitan el intercambio de información (Domínguez,2011).

En el ámbito educativo se ha visto la necesidad de crear espacios donde no hay tiempo o limitaciones de capacidad, la educación propone utilizar diversas plataformas para la enseñanza-aprendizaje (Díaz Forero, 2012).

En el contexto del desarrollo del caso cuasi experimental 4 se trabajó con estudiantes de la UTPL que cursaron la asignatura de Programación Web, con una carga horaria de 128 horas académicas. El curso con el cual se desarrolló el caso experimental estaba compuesto por 25 estudiantes que utilizaron y desarrollaron las diferentes actividades académicas relacionadas con la programación de aplicaciones en contextos reales, que fueron soportadas por las herramientas de computación en la nube y principios de la CA para la construcción de los prototipos propuestos en el marco del aprendizaje de la asignatura.

En la Figura 46 se puede apreciar el grupo de control que desarrolló el caso cuasiexperimental.

Figura 46
Grupo de Control



Nota. Grupo de control caso cuasi experimental 4

En el mismo contexto en el presente caso experimental se propuso la potenciación y valorización de la gastronomía en la ciudad de Cuenca, mediante el desarrollo de una aplicación web y móvil de carácter educativa basada en los principios de la CN y CA. El desarrollo de las aplicaciones de software antes mencionadas permitió el registro de los restaurantes para promocionar los negocios gastronómicos al público y comercializar sus platos por medio de publicaciones dentro de la aplicación, que cuenta con la información de cada plato: nombre del recurso gastronómico, principales ingredientes, categoría, precio, recursos multimedia interactivos donde prima el rescate cultural. Así mismo, los usuarios tienen acceso a la aplicación creando su propio perfil, bitácora. Mediante técnicas de inteligencia artificial, algoritmos de recomendación (SVD) y CA se generó una comunidad virtual de carácter culinario.

La implementación del aplicativo se dividió en partes denominadas iteraciones. Cada iteración consta de 4 fases: planificación, diseño, desarrollo y pruebas de funcionalidad. En la tabla 26 se presentan las fases antes mencionadas.

Tabla 26
Fases de las Iteraciones

Planificación: durante esta etapa se realizó una reunión para proponer ideas sobre cómo debe ser la funcionalidad de cada componente o producto entregable.
Diseño: durante esta etapa se realizaron diferentes backups del entregable para determinar la correcta posición, colores, tipografías de los entregables.
Desarrollo: durante esta etapa, se hizo uso de la técnica de programación en pares, uno de los integrantes trabajaba a la par con el resto de los programadores.
Pruebas: Una vez terminado el entregable se aplicaron una serie de pruebas para verificar que esté funcionando de la manera correcta.

Nota. Iteraciones de la metodología utilizada

Se utilizó la metodología de desarrollo Mobile-D que permitió interactuar constantemente al equipo de trabajo con el cliente y responder rápidamente a los cambios que se puedan presentar durante la etapa de desarrollo del proyecto, permitiendo la reducción de tiempos de producción. Está compuesta por 5 fases cuya descripción se presenta en la tabla 27.

Tabla 27
Fases de Mobile D

Fase 1: EXPLORACIÓN
En esta fase se realiza la definición del alcance del proyecto y su establecimiento con las funcionalidades a las que se quiere llegar.
Fase 2: INICIALIZACIÓN
Se preparan los planes para las siguientes fases y se establece el entorno técnico (incluyendo el entrenamiento del equipo de desarrollo).
Fase 3: PRODUCCIÓN

Primero se planificó la iteración de trabajo en términos de requisitos y tareas a realizar, luego se preparan las pruebas de la iteración. Las tareas se llevaron a cabo durante el día de trabajo, desarrollando e integrando el código con los repositorios existentes.

Durante el último día se lleva a cabo la integración del sistema (en caso de que estuvieran trabajando varios equipos de forma independiente) seguida de las pruebas de aceptación

Fase 4: ESTABILIZACIÓN

Se llevan a cabo las últimas acciones de integración para asegurar que el sistema completo funciona correctamente. Los desarrolladores realizan tareas similares a las que deben hacer en la fase de "producción", aunque en este caso todo el esfuerzo se dirige a la integración del sistema.

Fase 5: PRUEBAS

Tienen como meta la disponibilidad de una versión estable y plenamente funcional del sistema. El producto terminado e integrado se prueba con los requisitos solicitados por el cliente y se eliminan todos los defectos encontrados.

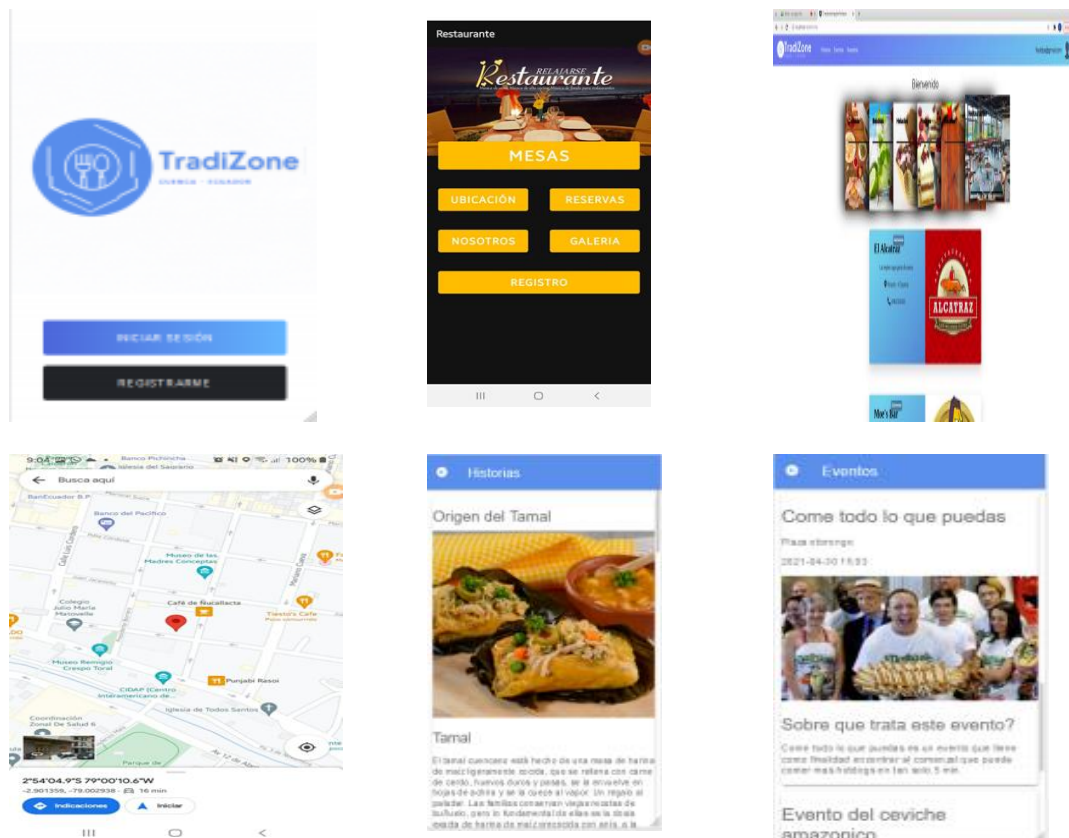
Nota. Descripción de las fases de la metodología Mobile D.

Resultados del caso cuasi experimental:

Este caso cuasi experimental incluye la creación de una aplicación web y móvil enfocada al rescate gastronómico de la ciudad de Cuenca, haciendo uso de Angular para el desarrollo del aplicativo web, Ionic para el aplicativo móvil y Firebase para el Backend.

En la Figura 47 se muestran las vistas de la aplicación móvil desarrollada.

Figura 47
Interfaces de Aplicación Móvil



Nota. Interfaces de Aplicación Móvil

La puesta en marcha de los prototipos generó el desarrollo de un turismo cultural de calidad, permitiendo el reaprovechamiento de los contenidos generados y facilitando su difusión por los nuevos canales y tecnologías. La principal característica de la página web, dentro de la gastronomía, es dar a conocer al mundo la exuberancia culinaria, mostrando comida desde la más exótica hasta la más común, las diferentes plataformas gastronómicas ayudan a difundir información generando a la vez impacto visual. En la Figura 48 se puede apreciar el prototipo inicial sitio web educativo desarrollado.

Figura 48
Sitio web desarrollado en el caso cuasi experimental

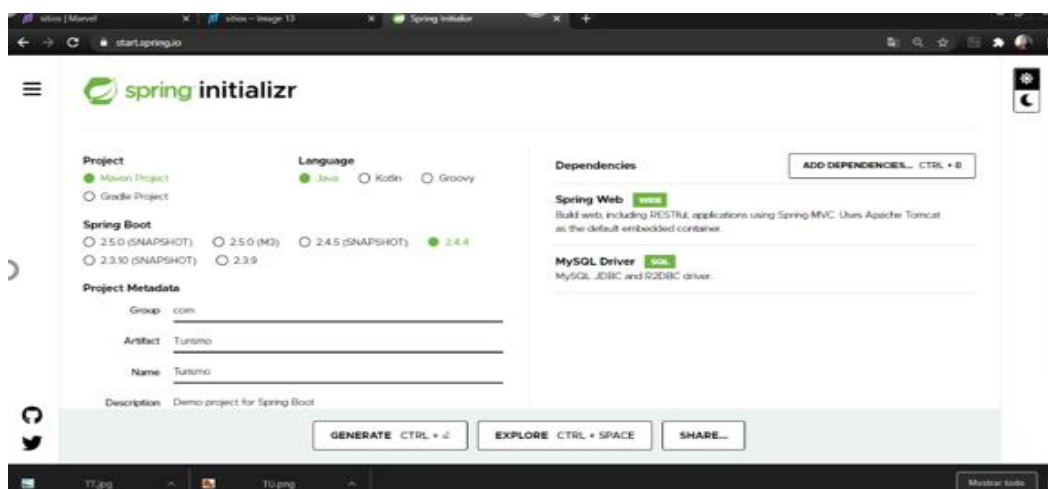


Nota. Aplicación Web desarrollada en el caso cuasiexperimental

Dentro del mismo orden de ideas, se detalla algunas características de la construcción de las aplicaciones antes mencionadas. La separación de los micros servicios permite realizar aplicaciones especializadas y fácilmente reutilizables, dan mucha potencia y flexibilidad a los sistemas de información, el automatismo de SpringBoot permite empezar a desarrollar prácticamente de inmediato y desplegar la aplicación sin complicaciones (Arteco, 2020).

En la Figura 49 se puede apreciar la configuración de Spring boot como ambiente para el desarrollo de las aplicaciones propuestas.

Figura 49
Configuración del ambiente de desarrollo



Nota. Configuración de ambiente para desarrollo

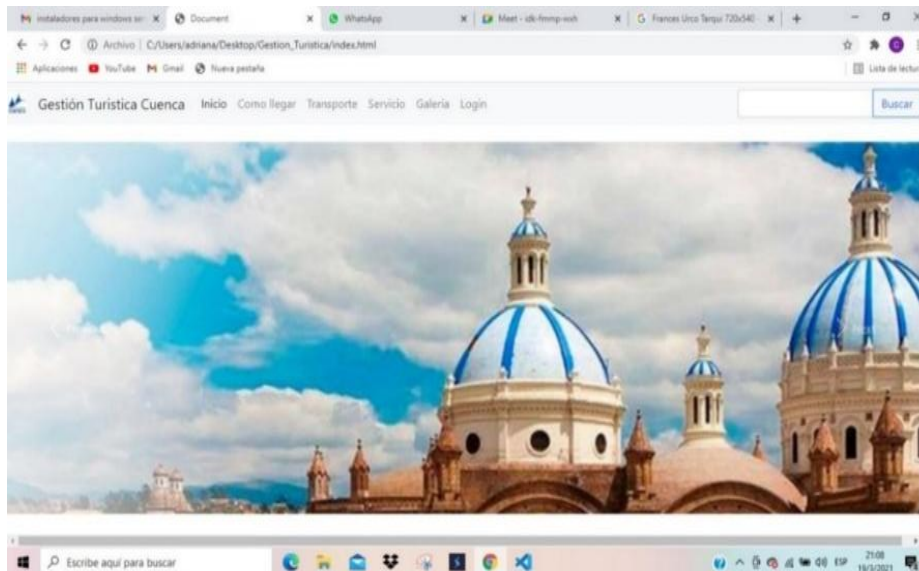
Luego de la configuración del ambiente, los estudiantes que desarrollaron el caso cuasi experimental 5 procedieron con creación REST en Spring Boot, con sus respectivos modelos. Seguidamente, configuraron los atributos especificando el nombre de tabla y columnas, con sus entity y las relaciones One to One, etc. En los controladores con cada entidad, aquí colocamos los métodos como el guardar, eliminar, listar, estos controllers llaman a su servicio correspondiente y desde este servicio se llama al repository JPA. Se lo creó con Java 11 y Jar, con las dependencias Swagger 2, versión 2.9.2.

De acuerdo con Menezes et al. (2022) los sistemas de software modernos se construyen comúnmente sobre marcos. Para acelerar el proceso de aprendizaje de las funciones proporcionadas por los marcos, se ponen a disposición ejemplos de código para ayudar a los desarrolladores. En el contexto del presente caso cuasi experimental 4 el marco SpringBoot es el componente central del sistema. Es responsable de recopilar, almacenar y analizar datos de los puntos de monitoreo de consumo de energía, además de recibir y procesar solicitudes de datos desde la capa de interfaz de visualización. Así mismo, Zhan et al. (2021) manifiestan que SpringBoot también proporciona interfaces de datos estándar para los programas externos.

La capa de la interfaz de visualización se desarrolla sobre la base del marco Vue y se integra en el marco Spring Boot. La capa de visualización combinó una biblioteca de gráficos de visualización de código abierto. Seguidamente se crearon las clases respectivas para generar la estructura del proyecto y los respectivos microservicios (Klymash et al.,2021). Es importante mencionar que el proyecto web fue desarrollo en Angular, versión 10, y para trabajarlo se utilizó el editor de texto Visual Studio Code, descrito en el capítulo 2 de la presente tesis doctoral. En la Figura 49 se puede apreciar la codificación del aplicativo web. Con base en los resultados obtenidos se llega a una conclusión acerca de la conveniencia de utilizar microservicios en la resolución de problemas aplicados. Luego del proceso de desarrollo de software, y del cumplimiento de las metodologías de desarrollo de software

previamente descritas, se obtuvo la aplicación, para luego ser publicada. En las Figura 50 se puede apreciar la aplicación previa a su publicación.

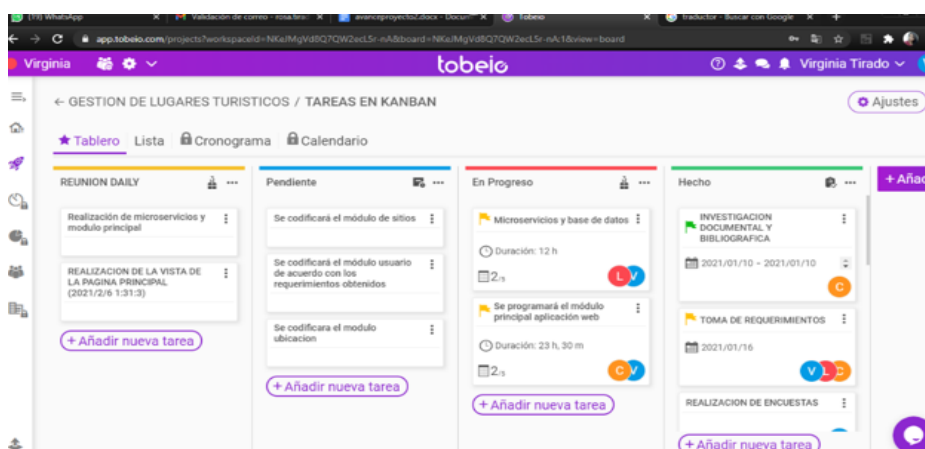
Figura 50
Interfaz de la aplicación desarrollada



Nota. Interfaz de la aplicación Web

Dentro del caso experimental se utilizaron como herramientas de CN el Tablero Kanban de Tóbelo, una herramienta ágil de gestión de proyectos diseñada para ayudar a visualizar el trabajo, limitar el trabajo en curso y maximizar la eficiencia (o el flujo). En la Figura 51 se puede apreciar la planificación de tareas del caso cuasi experimental desarrollado.

Figura 51
Tablero Kanban de Tobeio



Nota. Uso de herramienta Tobeio Tablero Kanban el caso cuasi experimental

6.1.5 Caso Cuasi experimental 5 Desarrollo de aplicativo web y móvil: tienda solidaria “Everything shop” (Universidad Nacional de Loja)

El desarrollo de aplicaciones web y móviles implica múltiples disciplinas y tecnologías, por lo que es necesario que los estudiantes adquieran las habilidades de programación en este tipo de plataformas. La enseñanza de la programación web es un campo amplio y dinámico que implica aprender diversos lenguajes de programación, herramientas y conceptos relacionados con el desarrollo de aplicaciones y sitios web.

Es importante considerar algunos aspectos en el contexto de la programación web.

Dentro del contexto del desarrollo de aplicaciones móviles es importante considerar los siguientes hallazgos en el presente caso cuasi experimental:

1. Lógica de implementación de aplicaciones fundamentada en una metodología de desarrollo de software orientada a este tipo de servicios.
2. Conocimiento de marcos de trabajo y bases de datos relacionales y no relacionales.
3. Diseño de la interfaz de usuario (UI) y experiencia de usuario (UX): Esto implica crear la estructura de la aplicación, el diseño visual, la navegación y la interacción con el usuario. Es crucial asegurarse de que la aplicación sea intuitiva y fácil de usar dentro del caso cuasi experimental desarrollado.
4. Desarrollo de la aplicación: aquí es donde los estudiantes crean el código de la aplicación utilizando lenguajes de programación adecuados, como Java o Kotlin para Android y Swift o Objective-C para iOS.
5. Pruebas y depuración: es esencial realizar pruebas exhaustivas para garantizar que la aplicación funcione correctamente y sea estable. Se deben detectar y corregir errores (bugs) y realizar pruebas de rendimiento en diferentes dispositivos y sistemas operativos móviles.
6. Implementación y distribución: una vez que la aplicación ha sido probada y se considera lista para su lanzamiento, se prepara para su distribución en las tiendas de

aplicaciones, como Google Play Store para Android o App Store para iOS. Se deben seguir las pautas y requisitos establecidos por cada plataforma.

7. Mantenimiento y actualizaciones: después del lanzamiento fue importante realizar un mantenimiento continuo de la aplicación, solucionar problemas, implementar actualizaciones y agregar nuevas características, según las necesidades y comentarios de los usuarios.

En el contexto del desarrollo del caso experimental 5 se trabajó con estudiantes de la Universidad Nacional de Loja que cursaron la asignatura de Proyectos Tecnológicos I (Enfoques de Programación) del séptimo ciclo de la carrera de Computación, con una carga horaria de 120 horas académicas.

El curso estaba compuesto por 20 estudiantes quienes utilizaron y desarrollaron las diferentes académicas relacionadas con la programación, soportadas por las herramientas de CN y principios de la CA, para la construcción de los prototipos propuestos en el marco del aprendizaje de la asignatura. En la Figura 52 se presenta el grupo de control que desarrolló el caso cuasi experimental 5.

Figura 52

Estudiantes participantes del grupo de control 5



Nota. Estudiantes participantes del grupo de control 5

El caso cuasi experimental tuvo como fin beneficiar a múltiples personas con la implementación de un software solidario, para que los interesados en ayudar a otros puedan

acceder, dar una contribución o aportar con algo. El sistema brindará apoyo a fundaciones para que por medio de él puedan recibir donaciones o adquirir productos.

Además, el sistema solidario posibilita que diferentes personas se hagan presentes constantemente y puedan interactuar con el responsable de supervisar al otro lado de la pantalla. Este software es en sí una tienda en la cual se podrán visualizar diferentes productos a un precio adquirible y beneficioso para ciertos establecimientos. Consideremos que hoy en día para estos lugares la forma más segura, confiable y rápida para generar compras es por medio de herramientas online que ofrecen estas cualidades y que actualmente son indispensables, es por esto por lo que las tiendas online son cada día más importantes para empresarios, comerciantes y para el público en general.

El caso cuasi experimental trató del desarrollo de una tienda online. Este sistema proporcionó un servicio para facilitar la comunicación y dar beneficios tanto a los propietarios de los negocios como a los consumidores. El sistema es de fácil manejo con una interfaz amigable para los usuarios, de manera que sus transacciones sean realizadas sin complicaciones.

Con el desarrollo de Everything Shop, los estudiantes proponen brindar un buen servicio a todos los usuarios que adquieran la aplicación web y móvil, de esta manera se estará satisfaciendo las diferentes necesidades que se presentarán con el pasar del tiempo, adicionalmente se evita que las personas viajen con sus equipos portátiles, lo que para la mayoría a veces representa mucha incomodidad o peligro, ya que en algunos lugares el índice de criminalidad es demasiado alto y lastimosamente a veces no se cuenta con seguridad. Uno de los principales objetivos es ofrecer comodidad al momento de buscar algún producto para vender o comprar, ya que la aplicación y sitio web tendrá elementos visuales para conocer mejor el producto, por lo tanto, se podrá escoger lo que esté de acuerdo al gusto o necesidad del cliente. Además, de brindar comodidad y seguridad se anhela estar acorde a las posibilidades económicas de cada interesado, debido a que en la aplicación y sitio web se mostrarán los precios reales de un producto. De esta manera cada usuario puede elegir lo

que más le convenga y cuando esté totalmente decidido podrá comunicarse con el proveedor, ya que dentro de la aplicación o del sitio web y de sus diferentes opciones encontrará el chat, de esta manera será más fácil llegar a un acuerdo para la compra y venta del producto seleccionado.

La finalidad de desarrollar una tienda solidaria virtual fue incrementar el número de donaciones o ventas, además que la empresa logre extenderse fuera de su ciudad y llegar a otras provincias, todo esto de una manera segura y económica, para que reciba más ayuda y crezca. Seguramente este proyecto servirá de aporte e incentivo para diferentes centros, ya que se pretende establecer las condiciones básicas necesarias que favorezcan al desarrollo de esta actividad y con ella la ayuda voluntaria de cada persona.

El funcionamiento de una tienda solidaria online durante las 24 horas del día fue algo novedoso para todas las personas, permitiéndoles de manera notable incrementar su confianza al realizar sus ventas o donaciones mediante la plataforma virtual. El presente proyecto aportó con nuevos aspectos en la herramienta a desarrollar, es decir ayudará acortando distancias, disminuyendo costos y ofreciendo todo tipo de productos para toda la empresa. Además, el dar a conocer esta nueva forma y perspectiva de Tienda Solidaria Online a la sociedad, nos dará como resultado una nueva visión de lo que será el crecimiento de la solidaridad.

Las metodologías empleadas para el desarrollo web y móvil del proyecto fueron las siguientes:

Metodología Lean. Una herramienta bastante utilizada y muy útil para varias personas en el mundo del software, cabe destacar que evita los desperdicios, ayuda a usar los recursos disponibles y sobre todo los que son imprescindibles, de esta manera los procesos mejoran de una forma sorprendente, así se pueden obtener resultados eficientes, evitar pérdidas en el transcurso del proceso, además de mejorar la calidad y reducción del tiempo (Martín-Conde

et al., 2021). El desarrollo de software bajo la filosofía Lean se puede resumir en siete principios presentados en la tabla 28.

Tabla 28
Principios de LEAN

Eliminar desperdicios/restos. Consideramos todo aquello que no aporta valor al cliente.

Amplificar el aprendizaje. Todos los miembros del equipo de desarrollo trabajan con una mentalidad de aprendizaje continuo.

Tomar decisiones lo más tarde posible. Este principio es considerado como primordial, ya que los requisitos de los clientes pueden cambiar de un día para otro, bien por modificaciones a las necesidades o por una mala definición de estas

Entregar lo antes posible. Las entregas de software son más frecuentes, incluyendo features alineadas con las user stories. Por este motivo, cada entrega incluirá funcionalidades que necesitan los usuarios lo antes posible, basadas en prioridades, impacto, valor o cualquier otro motivo.

Potenciar el equipo. Facilitar que los desarrolladores participen en la toma de decisiones de tiempos asociados a tareas y priorización de estas, además es importante que los miembros del equipo se sientan parte de él.

Visualizar todo el conjunto. Las interacciones del software creado con el resto de los sistemas dentro de la compañía, permitirá estudiar posibles mejoras y cambios que redunden en una mejor experiencia de usuario y aporten mayor valor para el cliente y para el equipo del proyecto.

Nota. Principios de LEAN aplicados en el caso empírico

De acuerdo con Almanza et al. (2012) no existe jerarquía entre los roles, pero podemos hablar de roles como tal y esto son presentados en la tabla 29.

Tabla 29
Roles LEAN

Instructional Designer.
Multimedia Developer.
Programmer.
Content Writer.
Editor.
Quality Assurance.
Graphic Artist.
Chief Engineer.

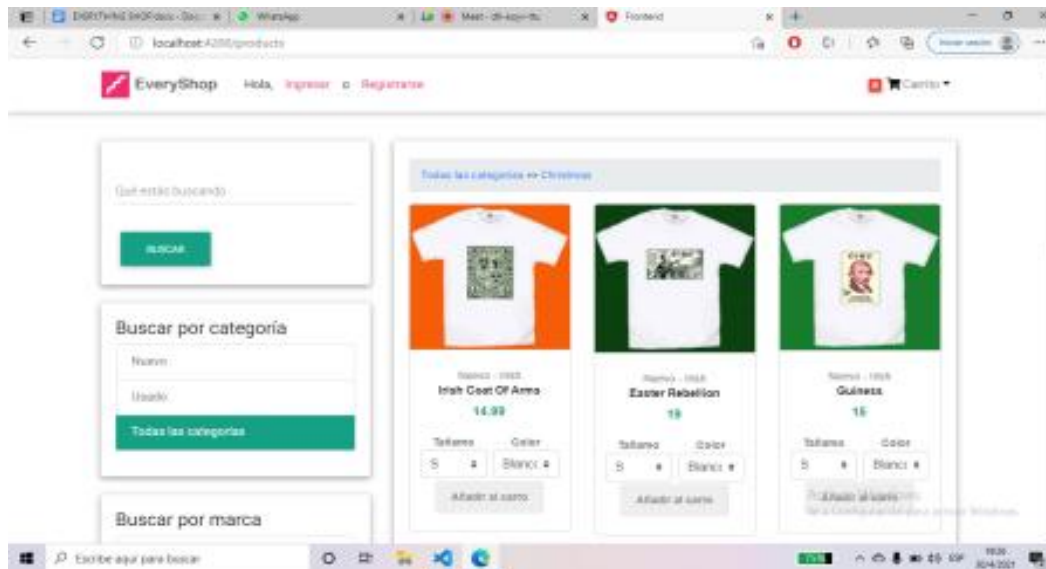
Nota. Roles LEAN

Según Garzas (2014) se considera desperdicio lo siguiente:

A continuación, se presenta el desarrollo de la aplicación web respectiva. La aplicación fue desarrollada en el framework Angular soportada con SpringBoot y complementada por herramientas de computación en la nube para la adecuada gestión de los procesos de desarrollo respectivos.

En este orden de ideas se presenta en la Figura 53 la interfaz de la aplicación web desarrollada en el presente caso experimental.

Figura 53
Aplicación Web Desarrollada



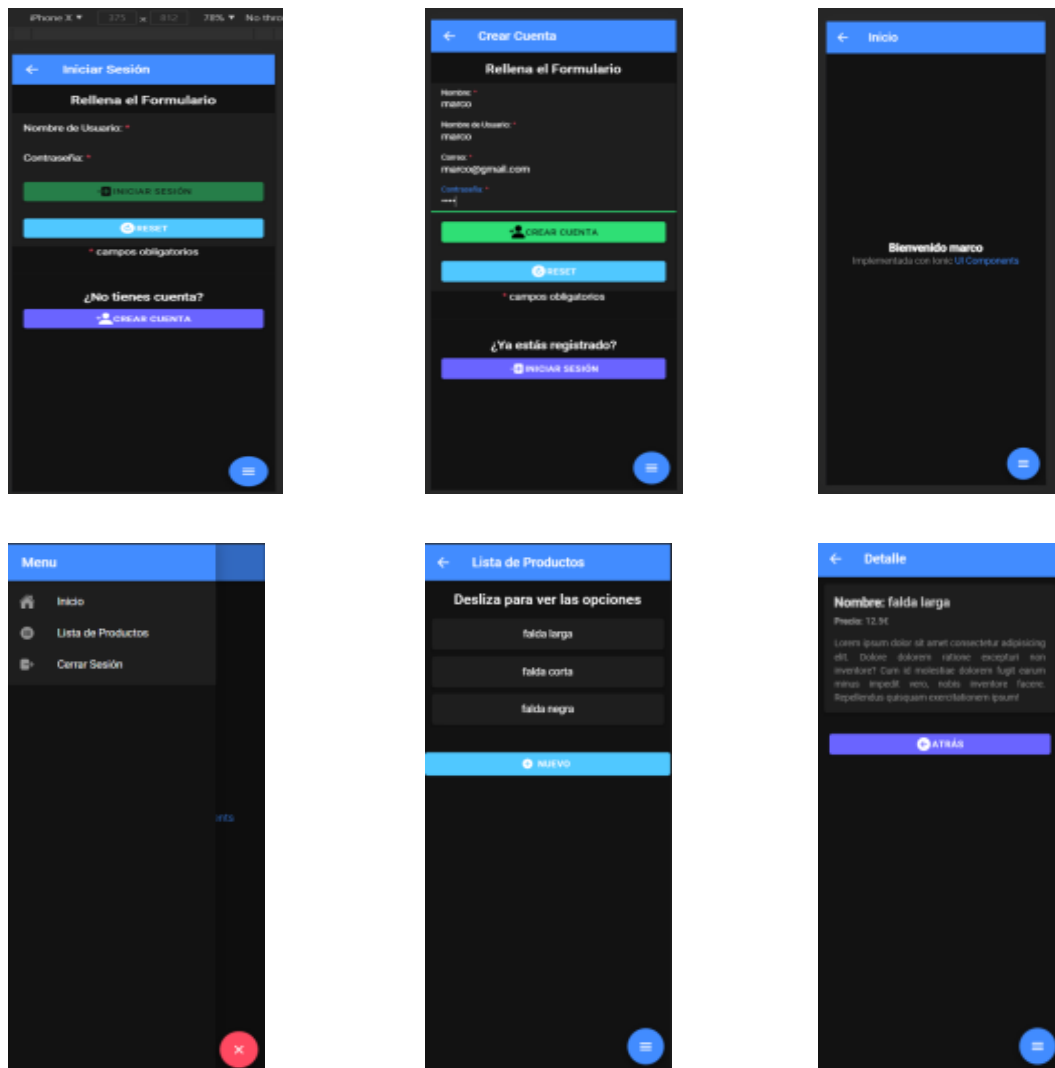
Nota. Aplicación Web Desarrollada

Los estudiantes del caso cuasi experimental desarrollaron una aplicación móvil, a continuación, se presentan algunas de las interfaces implementadas. Al ejecutar la aplicación lo primero que se le mostrará al usuario será el login, en caso de no estar registrado, procederá a hacerlo; luego se presentará la interfaz de bienvenida a la aplicación; posteriormente se presentará el menú de opciones al que tendrá acceso el cliente, en él encontrará las características del producto a adquirir.

En la Figura 54 se presentan las interfaces antes mencionadas.

Figura 54

Interfaces desarrolladas

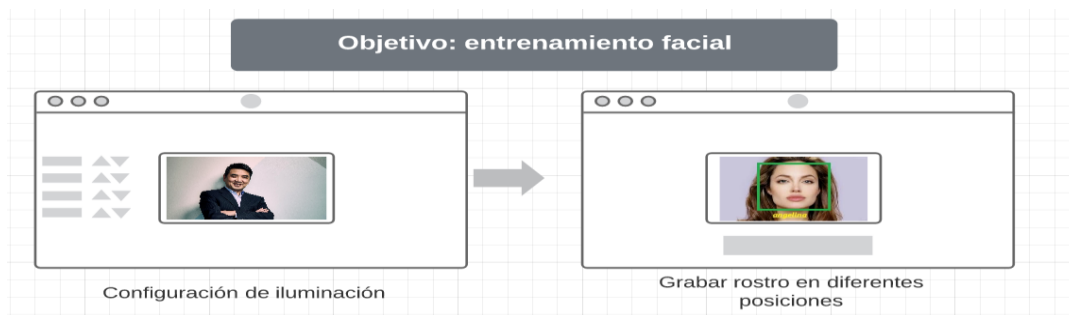


Nota. Interfaces de la aplicación móvil.

Dentro del mismo caso de estudio se desarrolló un prototipo para valorar la parte afectiva del comprador, el cual se ejecutaba luego de la adquisición, para medir el estado de ánimo. En la Figura 55 se presenta el prototipo para entrenamiento facial y grabación de los rostros, esto permitirá captar las emociones de cada usuario.

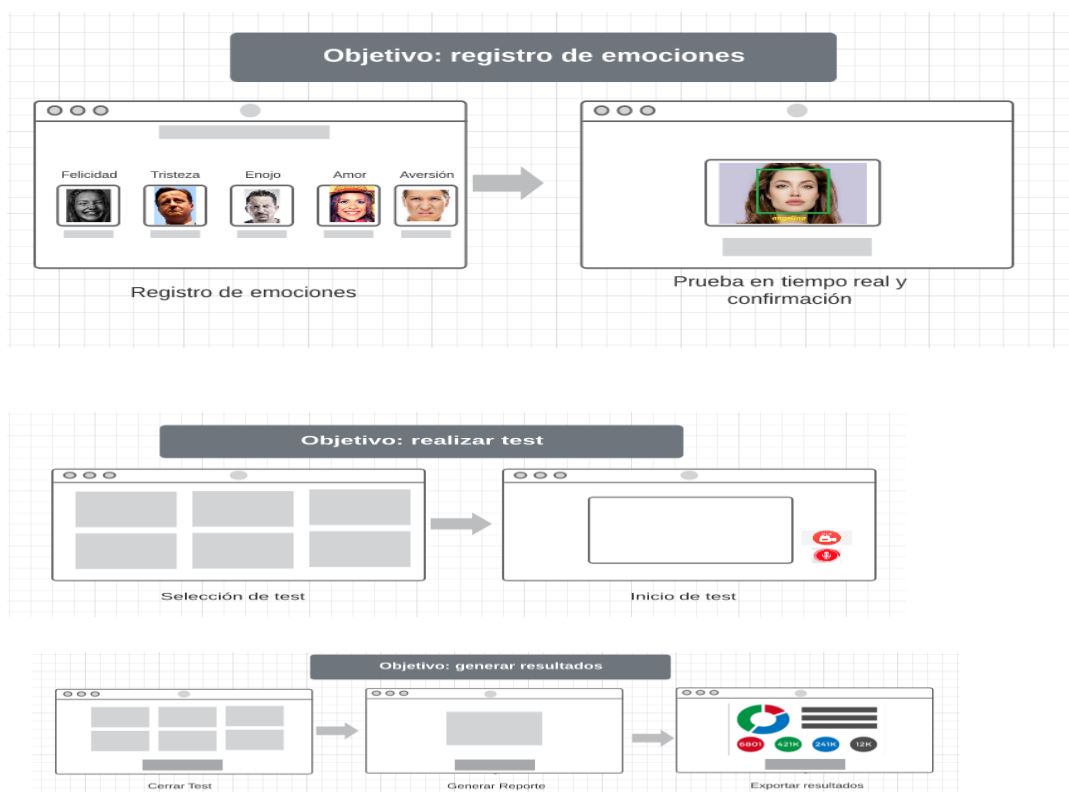
En la Figura 56 se genera el registro de emociones y la prueba y generación de resultados.

Figura 55
Prototipo Desarrollado



Nota. Prototipo de entrenamiento facial

Figura 56
Registro de emociones y generación de resultados de análisis emocional



Así mismo se utilizó JIRA, aplicación basada en web para el seguimiento de errores, de incidentes y para la gestión operativa de proyectos. Jira también se utilizó en áreas no técnicas para la administración de tareas relacionadas al desarrollo del proyecto. JIRA utiliza técnicas modernas como SMOTE, importancia de la permutación, junto con 7 modelos

diferentes de aprendizaje automático (Tagra et al.,2022) que permiten establecer algunos principios de CA. Jira también permitió expandir significativamente la funcionalidad básica con complementos personalizados (Kondratyev & Bondyakov,2021), esto facilitó el desarrollo de las habilidades computacionales y contextualizar en la realidad la aplicación de proyectos de software.

En este orden de ideas es importante mencionar que la minería de repositorios de software es un campo de estudio cuya tarea principal es extraer información valiosa de una gran cantidad de datos disponibles dentro de los repositorios de software (Valdez et al.,2020).

Otra de las herramientas utilizadas en el caso fue STORMBOARD, una fantástica herramienta 2.0 muy útil para realizar tormentas o lluvias de ideas entre varios usuarios, todo ello de forma online y con la posibilidad de añadir imágenes y otros elementos multimedia al tablero virtual donde se desarrolla la actividad. Con solo realizar el registro correspondiente se podrá crear tableros para las tormentas de ideas e invitar a otros usuarios añadiendo el correo electrónico.

Finalmente, se utilizó For my Innovation, un lugar colaborativo privado donde se puede almacenar y compartir información de modo seguro con un equipo de trabajo. Combina una red social con herramientas que le permitirán alcanzar los objetivos de un proyecto.

6.1.6 Caso Cuasi experimental 6: Desarrollo de Aplicativo Móvil para la Gestión de Restaurantes (Universidad Internacional del Ecuador)

En el contexto del desarrollo del caso cuasi experimental 6 se trabajó con estudiantes de la Universidad Internacional del Ecuador que cursaron la asignatura de Cliente Servidor 1 (Programación Móvil) con una carga horaria de 128 horas académicas.

El curso con el cual se desarrolló el caso experimental estaba compuesto por 15 estudiantes que utilizaron y desarrollaron diferentes destrezas académicas relacionadas con la programación de aplicaciones en contextos reales, que fueron soportadas por las herramientas de CN y por principios de la CA para la construcción de los prototipos

propuestos en el marco del aprendizaje de la asignatura. En la Figura 57 se puede apreciar el grupo de control que desarrollo el caso cuasiexperimental.

Figura 57
Grupo de Control



Nota. Estudiantes participantes en el caso cuasi experimental 6

Las aplicaciones de software desarrolladas se enfocaron en el contexto de la gastronomía; es por lo que el presente caso cuasi experimental consistió en la creación de una aplicación móvil de un restaurante, donde los clientes podrán hacer reservas a su gusto, así como también elegir el menú de una manera entretenida y nueva, así mismo podrán ver las promociones de la semana y hacer sus respectivos pedidos.

La aplicación móvil beneficiará tanto a las personas que se sirven del producto como a las encargadas de recibir y entregar los pedidos, ya que a través de la ella se realizará la transacción de manera rápida, evitando llamadas telefónicas, pérdida de tiempo, recepción errónea de pedidos e inconformidad del cliente, pues una vez que el cliente envíe su solicitud, a través de la aplicación, los datos serán captados en un sistema y el restaurante realizará el servicio de envío a domicilio.

Con la implementación de la aplicación móvil se verán reflejados los beneficios de los avances tecnológicos y cómo contribuyen al desarrollo de una empresa, ya que la aplicación proporcionará información actualizada para promocionar, comercializar productos y agilizar los pedidos a domicilio, aportando así al desarrollo tecnológico del restaurante, facilitando la

recepción de pedidos para los empleados y mejorando el servicio a los clientes dentro y fuera del restaurante.

Descripción del Proyecto

Registro de datos: el usuario deberá registrar los datos requeridos para acceder a la aplicación.

Ingreso a la aplicación: para que pueda acceder a la aplicación debe haber realizado su registro con el e-mail y contraseña.

Mostrar lista de productos: los productos deben cargarse en la aplicación y estar actualizados.

Mostrar información del restaurante: la aplicación móvil debe mostrar la información del restaurante: sucursales, dirección, teléfono, e-mail, etc.

Consultas: deben hacerse consultas por categorías y productos, evitando de esta forma que el usuario recorra la lista para encontrar un producto.

Lista de pedidos: la interfaz de la lista de pedidos debe brindar al usuario facilidades para disminuir y aumentar la cantidad del producto, así como también para eliminar un producto de la lista.

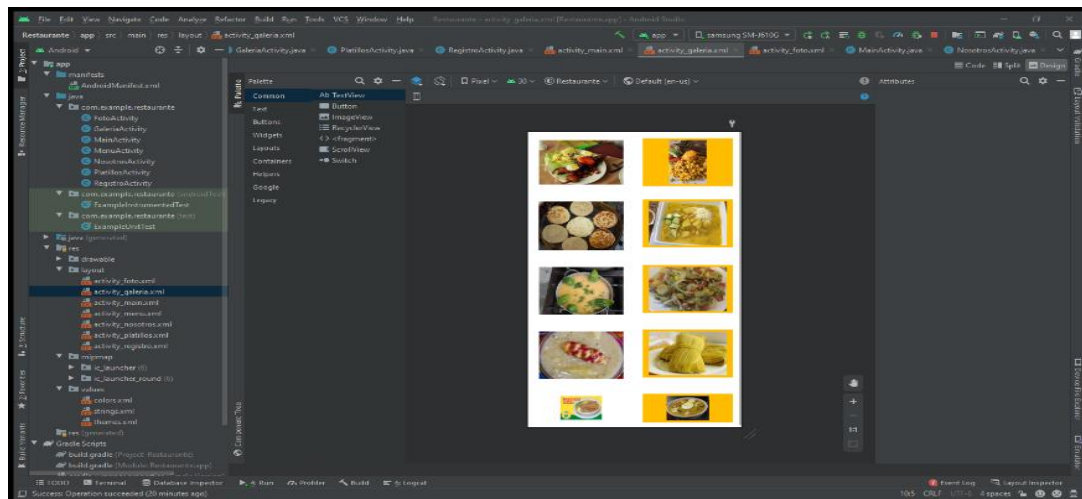
Transacciones de pedidos: con la lista de pedidos se debe ingresar la dirección del usuario, en caso de que la lista esté vacía o no cuente con la dirección no permitirá hacer el envío.

Verificar el estado del pedido: el usuario podrá conocer el estado en el que se encuentre el pedido del día, sea pendiente o atendido.

Interfaz amigable al usuario: el diseño de la aplicación debe ser sencillo y descifrable, para que permita la iteración del usuario y evitar inconvenientes al hacer uso de la aplicación.

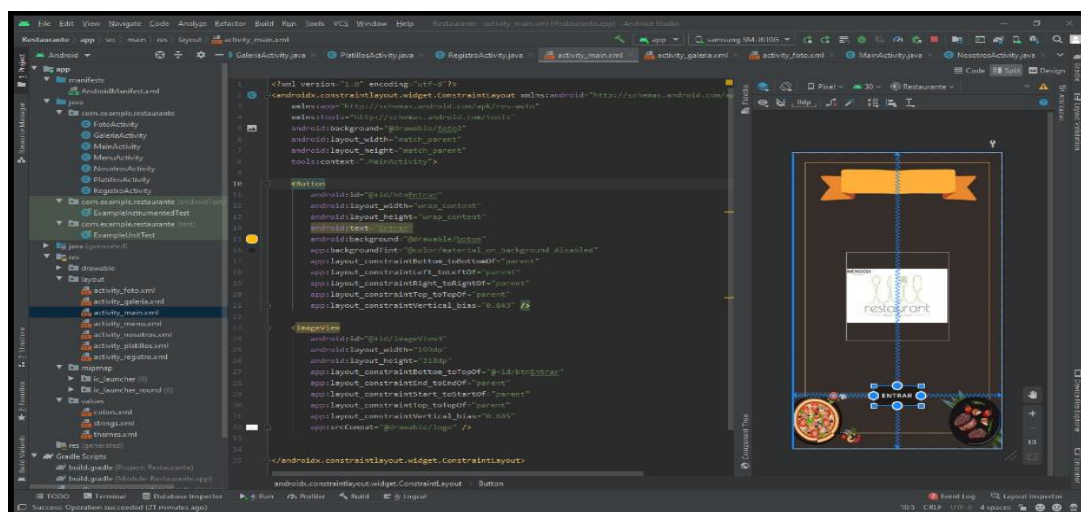
El aplicativo móvil fue desarrollado en la Plataforma Android Studio; en las Figuras 58 y 59 se puede apreciar la Interfaz de desarrollo de la aplicación antes mencionada.

Figura 58
Interfaz de la aplicación



Nota. Interfaz de Desarrollo de aplicación móvil

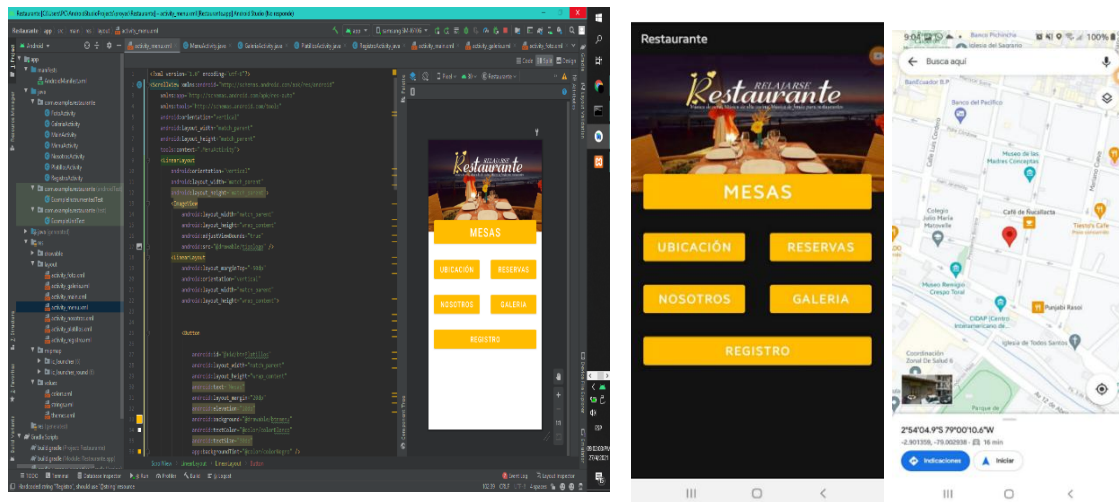
Figura 59
Interfaz de Desarrollo de la aplicación móvil



Nota. Interfaz de Desarrollo de la aplicación móvil

En la Figura 60 se puede observar la interfaz con los menús de opciones implementados, así como la implementación de Api de google maps para la localización de los restaurantes.

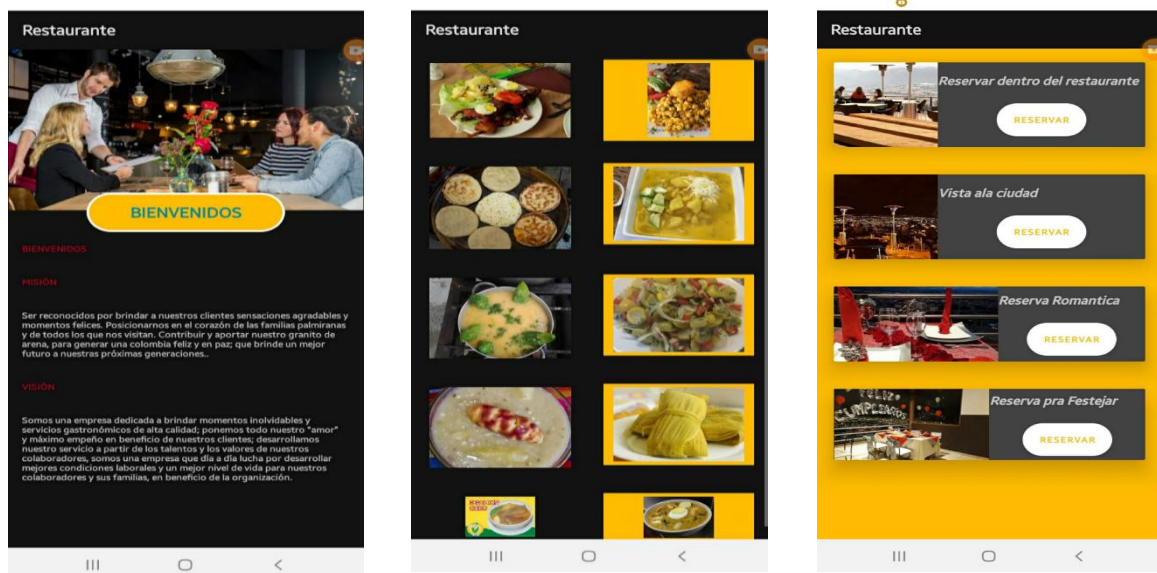
Figura 60
Menú del aplicativo y api de google maps



Nota. Menú del aplicativo y api de google maps

En la Figura 61, se presenta otras de las funcionalidades implementadas durante el desarrollo del aplicativo.

Figura 61
Funcionalidades implementadas



Nota. Funcionalidades implementadas

En el presente caso experimental se utilizaron las siguientes herramientas de CN:

AWS Cloud9.- Entorno de programación en la nube que permitió a los participantes del caso cuasi experimental escribir, ejecutar y depurar el código en el navegador. Como tal, incluye un editor de código, pero también un *debugger* y un acceso a terminal, lo que permitió validar ciertas funcionalidades del aplicativo, así como compartir, generar un trabajo colaborativo y con ello simular situaciones reales que se presentan en el campo laboral real.

Google Apps fue una de las Apps utilizadas en el presente caso empírico porque es uno de los mayores proveedores de servicios de computación en la nube, que ofrece múltiples instalaciones de colaboración. Las adaptaciones medibles y planificadas a Google Apps pueden proporcionar muchos beneficios para las instituciones de educación superior. Google Apps for Education (GAFE) es un servicio que proporciona varias versiones que se pueden personalizar con determinados productos de Google, en nuestro caso la Api de Google Maps.

Mahesh & Jayamma (2021) manifiestan que el desarrollo progresivo de la educación superior debe ser impulsado por la disposición a aclimatarse y nutrirse con la práctica de las tecnologías de la comunicación en todas las áreas de enseñanza, aprendizaje e investigación, siendo así que Google Apps for Education es una colección esencial de aplicaciones competentes que Google ofrece a todos los estudiantes e instituciones educativas de todo el mundo.

Es importante manifestar que las ciencias de la computación, en particular de la informática, es una de esas disciplinas que cobra sentido en la medida en que se articula a otras con el propósito de potenciar sus alcances y límites disciplinares (Fernández, 2006). Así mismo el desarrollo de estas nuevas competencias digitales plantea importantes retos a los sistemas educativos, toda vez que el modelo tradicional de instrucción no provee las estrategias requeridas por la sociedad actual en cuanto al desarrollo de habilidades para hallar y utilizar la información, construir significados e interactuar con diferentes culturas y contextos.

También se empleó REDMINE, una herramienta para la gestión de proyectos y el seguimiento de errores escrita usando el framework Ruby on Rails. Como base de datos soporta tanto MySQL como PostgreSQL o SQLite.

Finalmente, se usó Codeanywhere, un compilador de código configurado para dar soporte a todo tipo de lenguajes de programación.

Dentro de la presente investigación se propone un enfoque basado en marcos para el análisis de aprendizaje cognitivo sobre los conceptos y codificación que se enseñan en los cursos de programación, soportados con el uso de herramientas de CN y CA. El marco sirve como una plataforma que brinda estructura a los datos conceptuales, utilizando la técnica de mapeo conceptual y examina la propagación cognitiva de los alumnos sobre conceptos relacionados utilizando datos de evaluación.

El aprendizaje de la programación, en los casos cuasiexperimentales desarrollados, permitió a los estudiantes de informática adquirir las habilidades y conocimientos necesarios para escribir el código en un lenguaje de programación específico y utilizarlo para resolver problemas y crear programas de acuerdo con los requerimientos planteados. Existen muchos recursos disponibles para aprender a programar, desde libros y cursos en línea hasta tutoriales y comunidades en línea; pero el modelo de CN y CA, como herramientas en el desarrollo de las habilidades de programación, abre el camino para proponer nuevos modelos pedagógicos innovadores que aporten en el contexto educativo.

Las mejoras se evidenciaron a partir de la implementación de estrategias didácticas basadas en el uso de HCN y CA, las cuales fueron adaptadas según las características de cada asignatura de programación e integradas en los syllabus respectivos. Como resultado, se observó un progreso notable en las competencias de los estudiantes en relación con la programación.

Capítulo7: Desarrollo de modelo pedagógico

7.1 Pedagogía en la educación superior

La pedagogía es fundamental para redescubrir y redefinir la educación en cualquier momento histórico y sistema educativo especialmente en tiempos de cambios acelerados como el presente. De acuerdo a Suárez (2024), la pedagogía puede ayudarnos a redescubrir la educación en Internet. Sin embargo, cuando apelamos al potencial educativo de Internet, con mayor o menor intensidad, centramos nuestra atención educativa en saber qué tipo de aplicaciones tecnológicas pueden ser las más apropiadas para hacer eficaz, ameno o automático el proceso de aprendizaje (p.157).

La pedagogía de hoy, junto a sus temas clásicos, sus diversos enfoques y a sus cada vez más diversas disciplinas, debe afinar sus planteamientos y ser sensible al hecho de que las herramientas de CN y la CA no son solo parte del mobiliario didáctico, sino que su empleo -como entorno de aprendizaje y enseñanza- entraña el ejercicio de otra cultura educativa en especial en el ámbito de las ciencias de la computación e ingeniería en informática.

El propósito de la Pedagogía en Educación Superior (PES) es fomentar un aprendizaje profundo, crítico y autónomo, preparando a los estudiantes para ser profesionales competentes y ciudadanos informados; así mismo, la PES se refiere al estudio y aplicación de métodos y técnicas para mejorar el aprendizaje y la enseñanza en la educación postsecundaria. Esto incluye la planificación de cursos, la evaluación de estudiantes y la formación de maestros. La PES también se enfoca en desarrollar habilidades y competencias para el mundo laboral y en fomentar la investigación y el pensamiento crítico.

Uno de los objetivos de la educación superior es desarrollar habilidades y potenciarlas con el proceso pedagógico, desde una perspectiva científica y tecnológica, pretendiendo racionalizar los modelos de enseñanza, proponiendo como alternativa la descripción de los aprendizajes. El proceso educativo contemporáneo que es aplicado por el educador, y es considerado el más influyente, es la pedagogía activa que promueve la investigación antes que acumular teoría y contenidos.

Dentro de la pedagogía en educación superior es necesario establecer los objetivos de aprendizaje y las competencias a desarrollar en los estudiantes; así mismo se debe saber qué contenidos necesitan adquirir durante la formación. Otros aspectos considerados en la PES son:

- Las condiciones de agrupación de los estudiantes y la temporalidad planteadas para la formación.
- Saber y conocer bien, al inicio de la formación, quiénes serán nuestros estudiantes y qué nivel de conocimientos previos tienen.
- Saber con qué recursos contamos para llevar a cabo el trabajo.
- Decidir y comunicar cómo se evaluará el progreso de los estudiantes.
- Dominar como usuario las funcionalidades del entorno virtual. (Bautista et al., 2006).

A medida que la educación superior evoluciona y se diversifica, también lo hace la necesidad de métodos pedagógicos que atiendan las particularidades de este nivel educativo. La pedagogía en educación superior es esencial para garantizar que los estudiantes no solo adquieran conocimientos, sino también habilidades y competencias que les serán útiles en sus futuras carreras y vidas. Además, con el cambio constante en el mundo laboral y en la sociedad en general, la educación superior debe ser capaz de adaptarse y evolucionar para seguir siendo relevante y efectiva.

7.2 Modelo Pedagógico Propuesto

El cambio de un enfoque educativo tradicional a uno más activo, donde el estudiante juegue un papel protagónico en su aprendizaje, requiere tiempo y esfuerzo, lo cual puede ser difícil de encontrar en la rutina diaria. Diseñar experiencias y materiales que sean efectivos para los estudiantes, así como estrategias de uso adecuadas, puede resultar complicado, pero con el apoyo de las herramientas de CN y CA se puede mejorar la experiencia. El modelo pedagógico propuesto es un esquema representativo de una teoría educativa basada en el uso de herramientas de CN y CA; así mismo se consideran los aspectos curriculares de la

educación que aportan al área de las ciencias de la computación e informática. El modelo pedagógico propuesto especifica que, a la hora de diseñar cualquier situación instruccional o producto educativo, el docente debe pasar por una serie de fases, las que de acuerdo con este modelo son:

- Análisis.
- Diseño.
- Desarrollo.
- Implementación (Uso de las herramientas de CN y CA) en asignaturas cuyo componente principal sea la programación
- Evaluación.
- Lecciones Aprendidas.

Dentro del mismo orden de ideas es importante considerar ciertas interrogantes que nos den la perspectiva base para la puesta en marcha del modelo pedagógico propuesto.

- ¿Qué conocimientos previos pueden tener los estudiantes de esta acción formativa?
- ¿De qué recursos personales, temporales o materiales se dispone para desarrollar la acción formativa?
- ¿Qué conocimientos, habilidades o actitudes deben desarrollar los estudiantes a lo largo de la acción formativa?
- ¿Qué conocimientos, habilidades o actitudes deberán dominar los estudiantes una vez finalizada la acción formativa?

Es importante considerar los fundamentos de la práctica didáctica de los docentes en relación teoría- práctica educativa, las personas en su quehacer cotidiano y profesional, ven y actúan en el mundo a través de su propia teoría, las actitudes y aptitudes personales son nuestros lentes cognitivos que dirigen nuestra representación y actuación en el mundo. Desde una perspectiva científica y tecnológica, se pretende racionalizar los modelos de

enseñanza, proponiendo como alternativa la descripción de los aprendizajes esperados en términos de conductas observables y la programación exhaustiva de los medios (actividades, recursos, etc.) que las hacen posibles.

El papel del estudiante consistió en desarrollar un conjunto de respuestas de aprendizajes, siguiendo las pautas y secuencias de actividades determinadas por los docentes y que dentro de nuestro modelo se apoyan en el uso de las herramientas de computación en la nube y los principios de la CA. Así mismo, es importante considerar que la enseñanza y el aprendizaje son procesos interconectados que suponen el impulso fundamentado de cambios de conocimientos de los estudiantes, con base en el diseño de experiencias adecuadas, que en nuestro caso en particular fue desarrollar las destrezas de programación en los estudiantes del área de ciencias de computación e informática.

En el mismo orden de ideas, las estrategias de enseñanza son medios o recursos para prestar la ayuda pedagógica ajustada a las necesidades de progreso de la actividad constructiva de los alumnos, y de ello el uso de las herramientas de CN en los casos cuasi experimentales desarrollados en la presente tesis doctoral fueron uno de los pilares para la mejora de las destrezas de programación.

Las IES Ecuatorianas, en sus programas de formación en el área de ciencias de la computación, deben asumir el desafío de la modernidad y redefinir el entorno de la enseñanza de la programación, de manera que trascienda a los modelos tradicionales ya conocidos en este contexto, por lo que se debe adaptar a nuevos paradigmas que centren al estudiante en su aprendizaje, apoyado por el uso de herramientas de CN y CA. El modelo propuesto, en la presente tesis doctoral, surge de responder al cómo se puede innovar en la enseñanza de la programación y mejorar las destrezas de programación con el apoyo de herramientas complementarias en el proceso de enseñanza y aprendizaje, y que adicional a ello se soporte de una docencia dinámica donde el rol del docente sea orientador y genere un entorno adecuado de aprendizaje.

El modelo pedagógico propuesto se fundamenta en la relación estudiante y docente, está apoyado por el uso de las herramientas de CN y CA, estableciendo sus características y jerarquización. El modelo dinamiza los recursos tecnológicos que se requieren para la puesta en marcha del proceso respectivo y se estructura a partir de la definición de los siguientes componentes:

1. Propósito del modelo

Uno de los objetivos más importantes de la enseñanza de las ciencias de la computación (CC) e ingeniería en informática es fomentar el desarrollo de las habilidades computacionales, en especial de la programación, habilidades de pensamiento (HP) que son aplicables a muchas disciplinas educativas y áreas de la vida (Barr & Stephenson, 2011).

Papert (2020) aludió a las HP en su libro, en una discusión sobre el desafío de integrar la educación en Ciencias de la Computación (CC) a las experiencias cotidianas de los estudiantes. Más tarde, Wing popularizó el término y lo definió como un conjunto de habilidades de razonamiento para formular y resolver problemas usando computadoras y otras tecnologías de la información (Wing, 2011).

El propósito y finalidad de los procesos educativos en las IES, que ofertan programas de formación en el área de las ciencias de la computación, se orientan a formar estudiantes con sólidos conocimientos en la programación, que se adapten a las necesidades del mercado laboral; por ello mediante el modelo pedagógico propuesto se busca apoyar a los estudiantes a mejorar las destrezas de programación soportada por herramientas de CN y CA y que el conocimiento de estas herramientas permita una mejor adaptación laboral.

2. Contenidos

Los contenidos que se trabajan en cada IES, en el contexto de la enseñanza de la programación, deben ser disciplinares, culturales, sociales y no solo limitarse a la transmisión simple de información y al desarrollo de habilidades aisladas, deben centrarse con un creciente énfasis en nuevas formas de fomentar e inducir el conocimiento, de construcción y

de solución de problemas computacionales con el apoyo de herramientas computacionales, especialmente basadas en CN y soportadas por principios de la CA.

El enfoque en la enseñanza de la programación en las IES debe ser integral y multifacético e incorporar aspectos culturales y sociales. La enseñanza de la programación más allá de la simple transmisión de información y el desarrollo de habilidades aisladas debe centrarse en lo siguiente:

1. Fomentar e inducir el conocimiento: Los estudiantes deben ser motivados a explorar y comprender profundamente los principios y conceptos subyacentes de la programación y la computación.
2. Construcción de conocimiento: Los contenidos deben ser diseñados para ayudar a los estudiantes a construir conocimiento de manera activa, a través de la experimentación, la resolución de problemas y la colaboración.
3. Solución de problemas computacionales: Se debe poner énfasis creciente en enseñar a los estudiantes cómo abordar y resolver problemas complejos utilizando herramientas computacionales.
4. Uso de herramientas de CN: La integración de herramientas basadas en la nube es crucial, dado que estas ofrecen una gran flexibilidad, escalabilidad y acceso a recursos avanzados que pueden potenciar el aprendizaje y la práctica de la programación.
5. CA: Incorporar principios de la CA puede enriquecer la experiencia de aprendizaje, al considerar las emociones y el estado afectivo de los estudiantes en el proceso educativo. Esto puede ayudar a crear un entorno de aprendizaje más adaptativo y personalizado.

En resumen, la enseñanza de la programación en las IES debe evolucionar para incorporar una perspectiva más amplia y holística, que prepare a los estudiantes no solo con habilidades técnicas, sino también con la capacidad de aplicar estas competencias de manera creativa y efectiva en diversos contextos. Los sílabos en los que se introdujo el uso de las herramientas de CN buscan potenciar las habilidades de la programación de los estudiantes de las IES participantes y ser una base para estandarizar elementos y mejorar las habilidades de la programación.

3. Currículo

Es un proceso crucial para diseñar y organizar la estructura de enseñanza y aprendizaje en un entorno educativo, en nuestro contexto en el área de las ciencias de la computación e ingeniería informática. Un currículo bien desarrollado define los objetivos de aprendizaje, los contenidos a ser enseñados, los métodos de enseñanza y evaluación, y proporciona una guía clara para los docentes y alumnos.

La inserción de nuevos contenidos soportados por herramientas de CN son la base para una adecuada relación maestro-estudiante, lo que permite identificar la respuesta al cómo aprende el estudiante, ajustándose a su propio ritmo de aprendizaje e induciendo, de paso, su formación hacia la autonomía como desarrollador de software.

El uso de herramientas de CN facilita un entorno de aprendizaje adaptativo y colaborativo, donde los estudiantes pueden acceder a recursos actualizados y participar activamente en su proceso educativo.

4. Metodología

La enseñanza de la programación es un desafío que llama mucho la atención en el contexto educativo, así como en la creciente importancia de las habilidades digitales en el mundo moderno. Existen múltiples metodologías que se pueden adaptar según el grupo de aprendices y el contexto educativo. Entre las metodologías que se destacan para la enseñanza de la programación destacan:

1. Enfoque basado en proyectos: Los estudiantes aprenden a programar trabajando en proyectos concretos. Esta metodología fomenta la creatividad, la resolución de problemas y la aplicación práctica de los conceptos. En el contexto de la presente investigación se desarrollaron diferentes proyectos de software para solucionar un problema real, que fueron soportados con las metodologías de desarrollo de software y las herramientas de CN necesarias, según el proyecto.

Este enfoque no solo proporcionó una experiencia de aprendizaje más significativa, sino que también prepara a los estudiantes para enfrentar desafíos del mundo real en su futura carrera como desarrolladores de software.

2. Aprendizaje por descubrimiento: Permitir a los estudiantes explorar y experimentar con el código por sí mismos antes de ofrecer una explicación completa.

Beneficios del Aprendizaje por Descubrimiento en la Programación

1. Desarrollo de Habilidades de Resolución de Problemas: Los estudiantes aprenden a enfrentar y resolver problemas por sí mismos, lo cual es una habilidad esencial en la programación.
2. Mayor Retención del Conocimiento: Al descubrir y experimentar, los estudiantes tienden a recordar mejor los conceptos.
3. Fomenta la Curiosidad y la Motivación: El proceso de descubrimiento puede hacer que los estudiantes se sientan más involucrados y motivados para aprender.
4. Adaptabilidad y Flexibilidad: Los estudiantes aprenden a adaptarse a nuevas situaciones y a pensar de manera flexible, habilidades clave para los programadores.

Estrategias para Implementar el Aprendizaje por Descubrimiento dentro de la propuesta

1. Problemas y Proyectos Abiertos: Proporcionar a los estudiantes problemas abiertos o proyectos donde puedan explorar diferentes enfoques y soluciones.
2. Ambientes de Aprendizaje Interactivos: Utilizar entornos de programación interactivos donde los estudiantes puedan ver de inmediato los resultados de su código.
3. Guías y Pistas: En lugar de dar soluciones completas, ofrecer guías y pistas que orienten a los estudiantes en la dirección correcta.
4. Retroalimentación Constante: Proveer retroalimentación regular para ayudar a los estudiantes a refinar sus enfoques y comprender mejor los conceptos.
5. Fomentar la Colaboración: Promover el trabajo en grupo para que los estudiantes puedan aprender unos de otros y discutir sus hallazgos.

Programación en parejas: Dos estudiantes trabajan juntos en un solo ordenador, uno escribe el código (el conductor) mientras que el otro revisa y da sugerencias (el observador). Luego intercambian roles.

Esta metodología tiene varios beneficios:

Mejora la calidad del código: Al tener dos pares de ojos revisando el código constantemente, se pueden identificar y corregir errores más rápidamente, esto fue apreciado en el desarrollo de los casos cuasiexperimentales.

Aprendizaje compartido: Los estudiantes aprendieron de las fortalezas y conocimientos del otro, complementando sus habilidades y saberes.

Fomenta la comunicación y colaboración: Los estudiantes se comunicaron de manera efectiva para explicar y discutir sus ideas, lo que mejoró sus habilidades de trabajo en equipo.

Desarrollo de habilidades de revisión de código: Los estudiantes practicaron la revisión de código, una habilidad valiosa en el entorno profesional.

En resumen, la programación en parejas no solo mejoró la calidad del código, sino que también el aprendizaje de los estudiantes, además de fortalecer sus habilidades de trabajo en equipo y comunicación

3. Descomposición de problemas (Decomposition): Se enseñó a los estudiantes a dividir problemas complejos en tareas más pequeñas y manejables.
4. Práctica deliberada: Se animó a los estudiantes a practicar regularmente, resolver desafíos y retos de programación, y refactorizar el código para mejorar su eficiencia y legibilidad.
5. Enseñar patrones y antipatrones: Mientras que los patrones ofrecen soluciones a problemas comunes, los antipatrones muestran lo que no se debe hacer.
6. TDD (Test-Driven Development): aunque es una metodología más avanzada, enseñar a escribir pruebas antes del código puede ser una forma efectiva de enfocar a los estudiantes en la resolución de problemas y la calidad del código.
7. Aprendizaje basado en juegos: Utilizar juegos o plataformas gamificadas, como CodeCombat o CodinGame, para hacer el aprendizaje más interactivo y divertido.

8. Revisiones de código y feedback regular: Proporcionar retroalimentación constante a los estudiantes para que puedan identificar y corregir sus errores.
9. Integración de conceptos de otras disciplinas: Relacionar la programación con conceptos de matemáticas, ciencias, arte o cualquier otra disciplina relevante para hacerla más accesible y contextualizada.
10. Hacer hincapié en conceptos fundamentales: En lugar de centrarse en la sintaxis específica de un lenguaje, es esencial que los estudiantes comprendan los conceptos básicos, como variables, estructuras de control, estructuras de datos y algoritmos.
11. Utilizar recursos adicionales: Hay muchos recursos en línea, como tutoriales, cursos, foros y comunidades, que pueden complementar el aprendizaje.

Dentro del modelo planteado se estableció la combinación de HCN, metodologías de desarrollo agile, metodologías de desarrollo móvil, enfoque basado en proyectos (casos cuasi experimentales), aprendizaje por descubrimiento, constante retroalimentación de los docentes y estudiantes; así mismo se integró conceptos de otras disciplinas para generar productos de software de calidad en el marco de una simulación del desarrollo de productos software. Dependiendo de la asignatura de programación se realizó la adaptación de las HCN y la CA en los syllabus respectivos, obteniendo mejoras significativas en el desarrollo de las habilidades de la programación.

5. Recursos

Un recurso educativo es cualquier material, herramienta o medio que facilita el proceso de enseñanza y aprendizaje. Estos recursos pueden ser utilizados por educadores y alumnos para mejorar la comprensión de los contenidos, desarrollar habilidades y competencias, y enriquecer la experiencia educativa en nuestro contexto en el desarrollo de las habilidades de la programación. Los recursos didácticos y medios de apoyo al proceso de aprendizaje se conciben como soportes esenciales en el proceso de formación y en el modelo pedagógico, rompiendo estructuras tradicionales, por lo cual exigen una adecuada infraestructura tecnológica que sea soportada por las HCN y la CA.

En los capítulos 3 y 6 se describe un conjunto de herramientas de CN, así como los principios de CA utilizados para probar y evaluar el modelo propuesto. La generación de diferentes recursos de enseñanza y aprendizaje, en el contexto del desarrollo de las habilidades de la programación, fue uno de los pilares del modelo. La interacción de los estudiantes y maestros mediante herramientas de CN generó un adecuado ambiente de aprendizaje y de retroalimentación en el desarrollo de los diferentes prototipos creados en el marco de la presente investigación.

6. Evaluación

La evaluación se entiende como el proceso de investigación, obtención, y suministro de información útil y valorativa acerca del desempeño del estudiante, que en nuestro contexto se enfoca en la mejora de las habilidades de la programación. La evaluación de un modelo pedagógico es un proceso que permite analizar la eficacia, eficiencia, pertinencia y relevancia de un modelo de enseñanza y aprendizaje. Esta evaluación se lleva a cabo para identificar sus fortalezas y debilidades y, en función de ello, tomar decisiones que permitan mejorar la calidad educativa. Al evaluar un modelo pedagógico propuesto es importante considerar diversos aspectos:

1. **Objetivos y metas:** ¿Son claros y alcanzables? ¿Responden a las necesidades actuales de los estudiantes y de la sociedad?
2. **Contenido:** ¿Es relevante y pertinente? ¿Está actualizado y se ajusta a las necesidades del currículo de las ciencias de la computación?
3. **Metodología:** ¿Fomenta el aprendizaje activo, participativo y significativo? ¿Cómo se integran la tecnología y otros recursos? ¿Se considera la diversidad de estilos de aprendizaje de los estudiantes?
4. **Evaluación del aprendizaje:** ¿Cómo se evalúa a los estudiantes? ¿Se utiliza una variedad de instrumentos y técnicas? ¿Se promueve la autoevaluación y la coevaluación?

5. **Rol del docente:** ¿Qué papel juega el docente en el proceso de enseñanza-aprendizaje de la programación? ¿Es facilitador, guía, mediador o transmisor de información?
6. **Ambiente de aprendizaje:** ¿Es propicio para el aprendizaje? ¿Cómo se gestiona la relación entre pares, el trabajo en equipo y el clima de clase?
7. **Inclusión:** ¿El modelo contempla y da respuesta a las necesidades de todos los estudiantes, considerando diversidad cultural, habilidades, capacidades y ritmos de aprendizaje?
8. **Desarrollo profesional docente:** ¿Se ofrecen oportunidades para que los docentes se formen y actualicen?
9. **Resultados:** ¿Los estudiantes logran los objetivos de aprendizaje propuestos? ¿Cómo se compara el rendimiento de los estudiantes con otros modelos o estándares?
10. **Feedback y mejora continua:** ¿El modelo tiene mecanismos para recoger feedback de docentes, estudiantes y otros stakeholders? ¿Se realiza una revisión y mejora continua del modelo?

Cuando los estudiantes y docentes tienen acceso a herramientas colaborativas en línea, la dinámica del aprendizaje cambia. Se va más allá del aula tradicional y se crea un espacio en el que los participantes pueden colaborar en tiempo real, compartir recursos y trabajar en proyectos conjuntos, independientemente de dónde se encuentren físicamente. Este tipo de interacción no solo fomenta el trabajo en equipo y la colaboración, sino que también prepara a los estudiantes para el mundo laboral moderno, donde las herramientas de colaboración en línea y el trabajo remoto se han vuelto la norma.

7. Soporte Tecnológico del modelo pedagógico propuesto

El soporte tecnológico del modelo pedagógico se refiere a la integración de tecnologías, herramientas y plataformas para facilitar, potenciar o transformar los procesos de enseñanza y aprendizaje. A medida que la tecnología evoluciona, su uso en la educación

se diversifica, ofreciendo una amplia gama de posibilidades para adaptarse a diferentes enfoques pedagógicos. Dentro del modelo planteado se han considerado las diferentes herramientas de computación en la nube que facilitaron la enseñanza y desarrollo de las habilidades de la programación; las mismas que están alineadas a contenidos de las asignaturas en las cuales se probaron.

Es esencial recordar que la tecnología por sí sola no garantiza una enseñanza efectiva. La elección y aplicación adecuada de estas herramientas deben estar alineadas con los objetivos pedagógicos y ser implementadas de manera reflexiva y estratégica. Integrar tecnología en el modelo pedagógico no es simplemente adoptar la última herramienta o tendencia, sino hacer que esta tecnología pueda mejorar y enriquecer el proceso de enseñanza y aprendizaje.

Es importante mencionar que, al seleccionar una herramienta, es esencial tener en cuenta aspectos como la privacidad de los datos, las características de seguridad y la facilidad de uso para garantizar una experiencia educativa efectiva y segura; por ello se diseñó un modelo pedagógico que integre la CN y la CA para desarrollar habilidades de programación, es una idea innovadora que combina tecnología de vanguardia con una enseñanza personalizada y sensible a las emociones del estudiante.

Conclusiones

- La revisión sistemática de la literatura sobre el estado del arte en CN y CA en educación evidencia avances significativos en la aplicación de ambas tecnologías a los entornos de aprendizaje, aunque su integración aún presenta desafíos conceptuales, metodológicos y éticos. La CN se ha consolidado como una herramienta esencial para mejorar la accesibilidad y escalabilidad de los recursos educativos, facilitando entornos de programación remotos, la colaboración en tiempo real y el análisis de grandes volúmenes de datos para personalizar la enseñanza en nuestro caso para mejora de las habilidades de la programación en el área de las Ciencias de la Computación . Paralelamente, la CA ha evolucionado con el desarrollo de sistemas inteligentes capaces de detectar, interpretar y responder a las emociones de los estudiantes, optimizando la adaptación de los contenidos y la retroalimentación en función de sus estados afectivos. No obstante, la combinación de ambas tecnologías sigue en una etapa incipiente, con una limitada cantidad de estudios empíricos que validen su impacto conjunto en la enseñanza. Se han identificado brechas en la precisión del reconocimiento afectivo, la integración pedagógica efectiva y la privacidad en el manejo de datos sensibles, lo que subraya la necesidad de investigaciones más profundas y enfoques multidisciplinarios que permitan consolidar su aplicación en el ámbito educativo de ello la presente tesis doctoral aporta a integrar estos enfoques desde una mirada educativa.
- El uso de las herramientas de CN durante el desarrollo de la presente tesis doctoral fomentó un entorno de aprendizaje más interactivo entre docentes y estudiantes. Se lograron establecer redes de colaboración efectivas, lo que promovió una mayor participación e interacción entre los estudiantes esto es visible en las diferentes herramientas de CN utilizadas en los 6 casos cuasiexperimentales. Este entorno social colaborativo se reflejó claramente en los tiempos de respuesta y en el desarrollo ágil de los proyectos implementados. Además, la incorporación de nuevas tecnologías y paradigmas computacionales amplió significativamente el alcance del proceso de enseñanza y aprendizaje en programación de los estudiantes participantes en los casos

cuasiexperimentales. En este contexto, la aplicación de los principios de la CA permitió comprender mejor la influencia de las emociones en el aprendizaje, lo que demandó el desarrollo de habilidades computacionales específicas orientadas a la programación

- La CN nos ofreció beneficios como el acceso remoto a recursos educativos, la colaboración en tiempo real, la reducción de costos de infraestructura y la escalabilidad. Sin embargo, para que estas tecnologías sean adoptadas de manera efectiva, luego de la exploración empírica en este trabajo, se ha visto que es crucial abordar los problemas de seguridad y protección de datos. Aquí algunos puntos clave sobre este tema detectados en el desarrollo de la presente tesis doctoral:

Protección de Datos Personales: Las IES participantes en la presente investigación manejaron una gran cantidad de datos personales de estudiantes, profesores y personal administrativo. Es fundamental garantizar que esta información esté protegida contra accesos no autorizados y vulnerabilidades.

Cumplimiento Normativo: Las IES que formaron parte de la presente tesis doctoral cumplieron con diversas regulaciones sobre privacidad y protección de datos.

Cifrado de Datos: El cifrado de datos, tanto en tránsito como en reposo, fue una medida crítica para proteger la información sensible. Se tiene la seguridad de que, incluso si los datos son interceptados, no podrán ser leídos sin la clave de cifrado adecuada.

Control de Acceso y Autenticación: Se implementaron mecanismos robustos de autenticación, como la autenticación de dos factores (2FA), y se definieron claramente los roles y permisos de acceso, esto ayuda a minimizar el riesgo de acceso no autorizado.

Auditorías y Monitoreo Continuo: Desde la perspectiva docente se realizaron auditorías regulares y monitoreo continuo de la actividad en la nube, esto permitió detectar y responder rápidamente a cualquier incidente de seguridad.

La CN utilizada en la presente tesis doctoral permitió la flexibilidad y capacidad de adaptación debido a que las estudiantes pueden seguir ritmos distintos en su aprendizaje, además con el adecuado acompañamiento y guía del docente en el proceso de enseñanza y aprendizaje de la programación Web se contribuye a la mejora de las competencias y habilidades en los ámbitos educativos y profesionales.

- La CN y la CA permitieron la flexibilidad y capacidad de adaptación, debido a que los estudiantes pueden seguir ritmos distintos en su aprendizaje; además, con el adecuado acompañamiento y guía del docente, en el proceso de enseñanza y aprendizaje de la programación Web, se contribuirá a la mejora de las competencias y habilidades en los ámbitos educativos y profesionales.
- La implementación de los principios de la CA permitió comprender la influencia de las emociones en el proceso de adquisición de las destrezas computacionales enfocadas a la programación. En este contexto, la CA se refiere a la tecnología que puede reconocer, interpretar y responder a las emociones humanas. Al aplicar estos principios, se ha observado que las emociones juegan un papel crucial en el aprendizaje y desarrollo de habilidades en programación. Las emociones positivas, como el entusiasmo y la satisfacción, pueden aumentar la motivación y la persistencia, mientras que las emociones negativas, como la frustración y la ansiedad, pueden obstaculizar el aprendizaje y disminuir el rendimiento. Al comprender y abordar estos aspectos emocionales, se pueden diseñar entornos de aprendizaje más efectivos y personalizados que no solo se enfoquen en los aspectos técnicos de la programación, sino también en el bienestar emocional de los estudiantes. Esto puede incluir el desarrollo de herramientas y técnicas para reducir la ansiedad, incrementar la confianza y mantener la motivación alta durante el proceso de aprendizaje.
- La CA, en la presente investigación se enfocó en el reconocimiento y la respuesta que dan las máquinas a las emociones humanas, abrió nuevas puertas en el ámbito educativo,

particularmente en la enseñanza y aprendizaje de la programación. Integrar la CA en el aprendizaje de la programación demostró que el proceso de formación es multifacético, y que reconocer y atender a la dimensión emocional es tan crucial como la instrucción cognitiva. En resumen, la CA proporcionó una valiosa perspectiva sobre cómo las emociones influyen en el aprendizaje de la programación, permitiendo la creación de estrategias y herramientas más integrales y efectivas para la enseñanza de estas habilidades.

Se ha podido ver que las potenciales ventajas de la CN y la CA no se pueden desplegar si no se tiene un modelo pedagógico claro de trabajo, que solas las tecnologías no pueden hacer maravillas y que, en el caso de la formación en ciencias de la computación en Ecuador, se requiere de ideas sobre la finalidad, la metodología o la evaluación claras para que, de forma coherente la CN y la CA puedan aportar alguna experiencia positiva en la formación y desarrollo de habilidades blandas. Esta tesis deja claro que el diseño e implementación del modelo pedagógico basado en CN y CA representa un enfoque innovador que responde a las necesidades actuales de la educación digital. Desde una perspectiva educativa, este modelo no solo facilita el acceso a entornos de aprendizaje escalables y colaborativos a través de la nube, sino que también promueve un aprendizaje adaptativo mediante el reconocimiento y la respuesta a las emociones de los estudiantes que formaron parte de los seis casos cuasiexperimentales implementados.

Referencias

- Abeer-Yassin, A., Shidlovskiy, S. & Jamal, A. (2018). Development of expert systems for analyzing electronic documents. *Serie de conferencias IOP: Ciencia e ingeniería de materiales*, 363(1). doi:10.1088/1757-899X/363/1/012003
- Aguilera-Hintelholher, R.M. (2013). Identidad y diferenciación entre método y metodología. *Estudios políticos*, 9(28), 81-103. [https://doi.org/10.1016/S0185-1616\(13\)71440-9](https://doi.org/10.1016/S0185-1616(13)71440-9)
- Alarcón, P., Fernández, D., Mahillo, M., y Álvarez, M. (2014). Gestión automatizada de Tutorías. *REDU: Revista de Docencia Universitaria*, 12(2), 351–373. doi: <https://doi.org/10.4995/redu.2014.5653>
- Al-Samarraie, H., & Saeed, N. (2018). A systematic review of cloud computing tools for collaborative learning: Opportunities and challenges to the blended-learning environment. *Computers and education*, 124(4), 77-91. <https://doi.org/10.1016/j.compedu.2018.05.016>
- Alim, E. & Hai, J. (1-2 de noviembre de 2017). *Deployment of cloud computing for higher education using google apps* [Ponencia]. 2nd International Conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE). Yogyakarta, Indonesia, 73.77. doi: 10.1109/ICITISEE.2017.8285563
- Al-Shawwa, B. & Ingram, D. (2021). Enuresis and sleep fellowship education. *Sleep and Breathing*, 26(3), 1377-1380. doi: <https://doi.org/10.1007/s11325-021-02518-6>
- Al-Wesabi, F., Obayya, M., Hamza, M., Alzahrani, J., Gupta, D. & Kumar, S. (2022). Energy aware resource optimization using unified metaheuristic optimization algorithm allocation for cloud computing environment. *Sustainable Computing: Informatics and Systems*, 35(17). doi: 10.1016/j.suscom.2022.100686
- Aljamal, R., Jubair, F. & El-Mousa, A. (16-18 de november de 2021). *Analyzing the cost-effectiveness of running HPC applications on microsoft azure virtual machines* [Conference]. IEEE Jordan International Joint Conference on Electrical Engineering

- and Information Technology (JEEIT), Ammán, Jordania. doi: 10.1109/JEEIT53412.2021.9634127.
- Alamsyah, M., Marhento, G., Siburian, M., Astuti, I. & Bhakti, Y. (2021). Application of blended learning with edmodo based on POE learning model to increase students understanding of science concepts. *Journal of Physics: Conference Series*, 1806(1), 12121. doi: <https://doi.org/10.1088/1742-6596/1806/1/012121>
- Acevedo, A., Sánchez, D., y González, D. (2017). ¿Por qué calidad y no excelencia educativa? Una breve reflexión sobre la calidad educativa en Colombia. *Revista Latinoamericana de Ensayo y Opinión*. <https://hdl.handle.net/11323/7197>
- Amazon Web Services (AWS). (2020). *Amazon Programs for Research and Education*. <https://aws.amazon.com/grants/>
- Apelgren, K. & Giertz, B. (2010). *Pedagogical competences: A key to pedagogical development and quality in higher education*. Ryegard Uppsala University
- Arcaro, J. (1995). *Quality in Education: An Implementation Handbook*. Delray Beach: St. Lucie Press.
- Araya-Castillo, L. (2013). ¿Qué hemos aprendido sobre la calidad de servicio en educación superior? *Revista Pilquen - Sección Ciencias Sociales*, 2(16), 1-12. <https://www.redalyc.org/articulo.oa?id=347532061008>
- Aslakhanova, S., Markaryan, V. & Amadaev, A. (13 de diciembre de 2021). *Digital technologies as a tool for effective business management* [Conference]. Paper presented at the AIP Conference Proceedings, doi: <https://doi.org/10.1063/5.0075407>
- Askhoj, J., Nagamori, M. & Sugimoto, S. (8 de febrero de 2011). “*Archiving as a Service: A Model for the Provision of Shared Archiving Services Using Cloud Computing*.” [Conference]. Proceedings of the 2011 Conference, Seattle, WA. Japan, <http://dx.doi.org/10.1145/1940761.1940782>.

- Askhoj, J., Sugimoto, S. & Nagamori, M. (2011), "Preserving records in the cloud", *Records Management Journal*, Vol. 21(3), pp. 175-187.
<https://doi.org/10.1108/09565691111186858>
- Aranha, R., Corrêa C. & Nunes F. (2021). "Adapting Software with Affective Computing: A Systematic Review," in *IEEE Transactions on Affective Computing*, Vol.12(4), 883-899.doi: 10.1109/TAFFC.2019.2902379
- Avendaño C, William R. (2013). Un Modelo pedagógico para La Educación Ambiental Desde La Perspectiva De La Modificabilidad Estructural Cognitiva. *Revista Luna Azul*, (36),110-133.<https://www.redalyc.org/articulo.oa?id=321728584009>
- Avendaño-Castro, W. y Parada-Trujillo, A. (2011). Un modelo pedagógico para la reproducción y transformación cultural en las sociedades del conocimiento. *En: Investigación y Desarrollo*,19(2), 398-413. <http://www.scielo.org.co/pdf/indes/v19n2/v19n2a07.pdf>
- Awwad, A., Hoorani, A. Abo Mokh, A. (2022). The impact of employing googles apps in education on students' achievement, *Apuntes de clase sobre tecnología educativa. Springer, Singapur*. https://doi.org/10.1007/978-981-19-0101-0_7
- Baelo-Álvarez, R. (2009). *el e-learning, una respuesta educativa a las demandas de las sociedades del siglo XXI. Pixel-Bit. Revista de Medios y Educación*, (35),87-96.
<https://www.redalyc.org/articulo.oa?id=36812381007>
- Balladares, J., Avilés, M., y Pérez, H. (2016). Del pensamiento complejo al pensamiento computacional: retos para la educación contemporánea. *Sophie, Colección de Filosofía de la Educación* (21), 143-159. doi:10.17163/shop. n21.2016.06
- Bartolomé, A., y Grané, M. (2013). Interrogantes educativos desde la sociedad del conocimiento. *Aloma: Revista de Psicología, Ciències de l'Educació i de l'Esport*, 31(1), 73-81. <https://revistaaloma.blanquerna.edu/index.php/aloma/article/view/173>
- Bartolomé, A., y Gallego, M. (2019). *Technology at the university: achievements and failures. REDU. Revista de Docencia Universitaria*, 17(1), 9-13.
doi: <https://doi.org/10.4995/redu.2019.12016>

- Barr, D., Harrison, J. & Conery, L. (2011). *Computational thinking: A digital age skill for everyone. Learning and leading with technology*, 38(6), 20–23. <https://id.iste.org/docs/learning-and-leading-docs/march-2011-computational-thinking-ll386.pdf>.
- Barrouillet, P. & Lecas, J. (1999). *Mental models in conditional reasoning and working memory. Thinking & Reasoning*, 5(4), 289–302.doi: 10.1080/135467899393940
- Barreto, C., y Diaz, F. (2017). *Las Tic en la educación superior: Experiencia de innovación*. (1.^a ed.). Editorial Universidad del Norte. <https://editorial.uninorte.edu.co/gpd-las-tic-en-educacion-superior.html>
- Bautista, G., Borges, F., y Forés, A. (2006). *Didáctica universitaria en entornos virtuales de enseñanza-aprendizaje* (1era ed.).Ediciones Narcea.
- Benson, K., Dowsley, R. & Shacham. H., (26-27 de noviembre de 2011). *Do you know where your cloud files are*, Association for Computing Machinery. Association for Computing Machinery [Conference]. Association for Computing Machinery, Nueva York <http://dx.doi.org/10.1145/2046660.2046677>.
- Ben-Ammar, M., Neji, M. & Gouardères, G. (2020). The affective tutoring system expert systems with applications, *Sistemas expertos con aplicaciones*, 37(4), 3013–3023. <https://doi.org/10.1016/j.eswa.2009.09.031>
- beServices. (2018). *Cloud computing y servicios IT*. be Services. <https://bit.ly/3kAYRPk>
- Bitrix, Inc. (3 de junio de 2020). *Software de intranet social gratuito*. Bitrix 24 <https://www.bitrix24.es/uses/software-de-intranet-gratuito.php>
- Blanco, G., y Blanco, P. (2007). La medición de la calidad de servicios en la educación universitaria. *Cuaderno de Investigación en la Educación*, 22(3), 121-136. <https://revistas.upr.edu/index.php/educacion/article/view/13270>
- Bora, U. & Ahmed, M. (2013). E-learning using cloud computing. *International Journal of Science and Modern Engineering*, 1(2), 9-12. <https://www.ijisme.org/wp-content/uploads/papers/v1i2/B0111011213.pdf>

- Brailovsky, D. (2018). *Lo nuevo y lo tradicional en educación: una oposición engañosa*. *Revista Senderos Pedagógicos*, 9(9), 161–178. <https://doi.org/10.53995/sp.v9i9.963>
- Brush, T., Glazewski, K. & Hew, K. F. (2008). Development of an instrument to measure preservice teachers' technology skills, technology beliefs, and technology barriers. *Computers in the Schools*, 25(1), 112-125. <http://dx.doi.org/10.1080/07380560802157972>
- Besker, T., Martini, A., & Bosch, J. (2018). Managing architectural technical debt: A unified model and systematic literature review. *Journal of Systems and Software*, 135(12), 1-16. doi: 10.1016/j.jss.2017.09.025
- Brereton, P., Kitchenham, B., Turner, M. & Khalil, M. (2007). Lessons from applying the systematic literature review process within the software engineering domain. *Journal of Systems and Software*, 80(4), 571-583. doi: <https://doi.org/10.1016/j.jss.2017.09.025>
- Brown, AL (1992). Experimentos de diseño: desafíos teóricos y metodológicos en la creación de intervenciones complejas en entornos de aula. *Journal of the Learning Sciences*, 2 (2), 141–178. doi: https://doi.org/10.1207/s15327809jls0202_2
- Cabero, J. (2014). Nuevas miradas sobre las TIC aplicadas en la educación, *Revista Andalucía Educativa*, 81(3),13-25. <https://www.redalyc.org/articulo.oa?id=15800304>
- Cabero, J., Marín, V., y Castaño, C. (2015). Validación de la aplicación del modelo TPACK para la formación del profesorado en TIC. *@tic. Revista d'innovació educativa* (14), 13-22. doi:10.7203/attic.14.4001
- Cabero-Almenara, J., Gutierrez-Castillo, J., Guillen-Gámez, F. & Gaete, A.(2022). Competencias digitales de estudiantes universitarios: creación de un modelo causal desde un enfoque PLS-SEM digital competencies of university students: creation of a causal model from a PLS-SEM approach, *Campus Virtuales*,11(1), 167-179. doi:10.54988/cv.2022.1.1008

- Cabero-Almenara, J., y Llorente-Cejudo, C. (2020). Covid-19: transformación radical de la digitalización en las instituciones universitarias. *Campus Virtuales*,9(2), pp. 25-34.
<https://dialnet.unirioja.es/servlet/articulo?codigo=8005978>
- Cabot, J. (23 de noviembre de 2018). *Editores de código online – programa sin salir de tu navegador*. Ingeniería de software. <https://bit.ly/3zgqkhq>
- Calvo, R., & D'Mello, S. (2010). Affect detection: an interdisciplinary review of models, methods, and their applications. *IEEE Transactions On affective computing*, 1(1), 18-37.doi: 10.1109/T-AFFC.2010.1.
- Casacuberta, D. & Vallverdú, J. (2010). Emociones sintéticas, *Páginas de Filosofía*,11(13), 116-144. <https://revela.uncoma.edu.ar/index.php/filosofia/article/view/121>
- Cantor, J., y Altavaz, A. (2019). Los modelos pedagógicos contemporáneos y su influencia en el modo de actuación profesional pedagógico. *Varona. Revista Científico Metodológica*,18(68),114-121 <http://scielo.sld.cu/pdf/vrcm/n68/1992-8238-vrcm-68-e19.pdf>
- Cambria, E. (2016) Affective computing and sentiment analysis, in *IEEE Intelligent Systems*, 31(2).102-107.doi:10.1109/MIS.2016.31
- Carpio-Jiménez, L., Suing, A. & Puertas-Hidalgo, R. (2022). Ecuador and the virtual high school education in pandemic times, *Smart Innovation*, 3(17),78-92. doi:10.1007/978-981-16-5063-5_25 p
- Cammariere, G., Portelli, M. & Russo, P. (04 de mayo de 2016). *GitHub analyser: A tool detecting class correlations on git repositories* [Conference]. Paper presented at the CEUR Workshop, Proceedings, Alemania. doi: 10.1730 35-40
- Cicciarella, S., Napoli, C.& Tramontana, E. (2015). Searching design patterns fast by using tree traversals. *International Journal of Electronics and Telecommunications*, 61(4), 321-326. doi:10.1515/eletel-2015-0041
- CloudShare. (2017). *Cloud Based Training Platform Y Test Labs*. CloudShare
<https://www.cloudshare.com/>

- Clough, S. M. (2015). Cloud computing: its concept and applications in libraries and information centers.2(5), 212-222. <https://www.ijtrd.com/papers/IJTRD8424.pdf>
- Chang, Y., Li, J. & Huang, D. (2022). A personalized learning service compatible with moodle E-learning management system. *Applied Sciences (Switzerland)*, 12(7), 3562-3577.doi:10.3390/app12073562
- Chakravarthi, K., Neelakantan, P., Shyamala, L., & Vaidehi, V. (2022). Reliable budget aware workflow scheduling strategy on multi-cloud environment. *Cluster Computing*, 25(2),1189-1205.doi:10.1007/s10586-021-03464-4
- Colina, L. (2007). La investigación en la educación superior y su aplicabilidad social. *Laurus*,13(25),330-353. <https://www.redalyc.org/articulo.oa?id=76111479016>
- Coll, C., Pozo, J. Sarabia, B. y Valls E. (1994). *Los contenidos en la Reforma. Enseñanza y aprendizaje de conceptos, procedimientos y actitudes* (1era ed.). Santillana. p. 202
- Coll, T. y Onrubia. J.(1016). Análisis y resolución de casos-problema mediante el aprendizaje colaborativo. *Revista de Universidad y Sociedad del Conocimiento*.3(2),pp.29-39.http://www.uoc.edu/rusc/3/2/dt/esp/coll_mauri_onrubia.pdf
- Crespo, M., y Palaguachi, M. (2020). Educación con Tecnología en una Pandemia: Breve Análisis. *Revista Cientific*,5(17), 292-310. doi: <https://doi.org/10.29394/Scientific.issn.2542-2987.2020.5.17.16.292-310>
- Croxall, B. (11 de septiembre de 2012)¿Pedagogía digital? Una desconferencia de pedagogía digital.Digitalpedagogy. <https://digitalpedagogy.hcommons.org/>
- Cooperberg, A.(2002). Las herramientas que facilitan la comunicación y el proceso de enseñanza-aprendizaje en los entornos de educación a distancia. RED. *Revista de Educación a Distancia*, 1(3),233-256. <https://www.redalyc.org/articulo.oa?id=54700302>
- Chen,S., Liu, J., Ma, F. & Huang, H. (2022). Customer-satisfaction-aware and deadline-constrained profit maximization problem in cloud computing. *Journal of Parallel and Distributed Computing*, 163(7), 198-213. doi: 10.1016/j.jpdc.2022.02.003

- Costagliola, G., De Rosa, M., Fuccella, V. & Ricchiuti, A. (2016). A software system in support of quality of degree programmes. *Proceedings of the 8th International Conference on Computer Supported Education*, 1(1),413-420. doi:10.5220/0005859504130420
- Dabbish, L., Stuart, C., Tsay, J. & Herbsleb, J. (12 de febrero de 2012). *Social coding in GitHub: Transparency and collaboration in an open software repository* [Conference]. Paper presented at the Proceedings of the ACM Conference on Computer Supported Cooperative Work, CSCW New York, 1277-1286. doi:10.1145/2145204.2145396
- De Benito, B. y Salinas, J. (2016). La investigación basada en diseño en Tecnología Educativa. *RIITE. Revista Interuniversitaria de Investigación en Tecnología Educativa*, 0, 44-59. Doi: <http://dx.doi.org/10.6018/riite/2016/260631>
- D'Mello, S. (2014). Confusion can be beneficial for Learning. *Learning and Instruction*, 29(4), 153–170. doi: <https://doi.org/10.1016/j.learninstruc.2012.05.003>
- Dekson, D., Suresh, E. & Ponnusamy, R. (10 de septiembre de 2009). *Intelligent system to teach programming languages* [Conference]. International Conference on Intelligent Agent & Multi-Agent Systems, Chennai, 1-5. doi: 10.1016/j.jpdc.2022.02.003
- Dhouib, S., y Halima, R. (18 de junio, 2013). Surveying collaborative and content management platforms for Enterprise [Workshop]. Infrastructure for collaborative enterprises, WETICE, 299-304. doi:10.1109/WETICE.2013.61
- D'Mello, S., Craig, S., Kaliouby, R., Alsmeyer, M. & Rebolledo, G. (09-10 de noviembre de 2007). *Modeling and scaffolding affective experiences to Impact learning* [Conference]. The 13th International Conference on Artificial Intelligence in Education, California, <https://openreview.net/forum?id=vWE70M3QkR>
- De Pablos Pons, J., Bravo, M. P. C. & Ramírez, T. G. (2011). Higher education supported through virtual platforms. changes in teaching practices: The case of the university of seville. *Estudios Sobre Educación*, 20(62), 23-48. <https://doi.org/10.20511/pyr2020.v8nSPE2.684>

- De Pablos, J., Colás, M., López Gracia, A., & García-Lázaro, I. (2019). Uses of digital platforms in Higher Education from the perspectives of the educational research. *REDU. Revista de Docencia Universitaria*, 17(1), 59-72. <https://doi.org/10.20511/pyr2020.v8nSPE>
- Del Prado, A., y Lamas, N. (2014). Alternativas para la enseñanza de pseudocódigo y diagrama de flujo. *Revista Electrónica Iberoamericana de Educación en Ciencias y Tecnología (RIECyT)*, 5(3), 102-113. 3.doi:<https://doi.org/10.54988/cv.2022.1.809>
- De Zubiria J. (2003). *Modelos Educativos Pedagógicos y Didácticos* (2da. Ed.).Ediciones SEM.
- De Zubiría, J. (2007). *Tratado de pedagogía conceptual: Los modelos pedagógicos*. Fondo de publicaciones Bernardo Herrera Meriño.
- Díaz, J.(2012). Simulación en entornos virtuales, una estrategia para alcanzar “Aprendizaje Total”, en la formación técnica y profesional. *Revista latinoamericana de estudios educativos México*,13(2),49–94.<https://www.redalyc.org/pdf/270/27024538003.pdf>
- Díaz, J. (2012). Modelos pedagógicos en educación a distancia. *Revista electrónica de humanidades, educación y comunicación social*,12(7),86-113, <https://doi.org/10.6018/red.557441>
- Doucet, A., Netolicky, D., Timmers, K. & Tuscano F. (2020). *Thinking about pedagogy in an unfolding pandemic*. An Independent report on approaches to distance learning during the COVID19 school closures. Inform the work of education International and UNESCO. <https://teachertaskforce.org/knowledge-hub/thinking-about-pedagogy-unfolding-pandemic>
- Dolog, P., Thomsen, L. & Thomsen, B. (2016). Assessing problem-based learning in a software engineering curriculum using bloom’s taxonomy and the IEEE software engineering body of knowledge, *Transacciones ACM sobre educación en computación (TOCE)*,16(3),1-41. doi: <https://dl.acm.org/doi/10.1145/2845091>
- Domínguez, G., Francisco, F., y López, E. (2011). *Orientación Educativa y Tecnologías de la Información y la Comunicación*(1ª ed.).Eduforma

- Dougiamas M. & Taylor P. (5 de marzo de 2003). *Moodle: using learning communities to create and open-source course management system* [Conference]. Proceeding of world conference on educational multimedia, hypermedia and telecommunication, Hawaii, 171-178. <http://editlib.org/p/13739/>
- De Vicente, A. (2003). *Towards Tutoring Systems that Detect Students' Motivation: An investigation*. [Thesis Doctoral, University of Edinburgh, Edimburgo, Reino Unido]. <https://www.ed.ac.uk/studying/postgraduate/applying/research/phds>
- Dybå, T., Dingsøyr, L. & Strength, T. (09- 10 de octubre de 2008). *Of evidence in systematic reviews in software engineering* [Conference]. In Proceedings of the second ACM-IEEE international symposium on empirical software engineering and measurement. (ESEM '08). Association for computing machinery, New York, 178–187. <https://doi.org/10.1145/1414004.1414034>
- EcuRed. (30 de mayo de 2020). DotProject, EduRed, <https://www.ecured.cu/DotProject>
- Elliott, C. (2002). *The role of elegance in emotion and personality: Reasoning for believable agents, Emotions in Humans and Artifacts*, 2002(34), 251-288. doi:10.7551/mitpress/2705.003.0009
- Fahmideh, M., Grundy, J., Beydoun, G., Zowghi, D., Susilo, W. & Mougouei, D. (2022). A model-driven approach to reengineering processes in cloud computing. *Information and Software Technology*, 1(144), 333- 352. doi: 10.1016/j.infsof.2021.106795
- Fernández, J. (21 de febrero de 2006). *Un científico bueno contribuye a los avances más que muchos malos*. Tribuna Complutense, <https://tribuna.ucm.es/news/autor/jaime-fernandez-1?gap=7>
- Feuerstein, R. & Rand, Y. (1974); Mediated Learning Experiences: An out-line of the proximal etiology for differential development of cognitive functions. *International Understanding, Educare*, 9(10), 7-36. doi:
- Flórez, R. (1999). *Evaluación Pedagógica y Cognición*. McGraw-Hill.

- Floridi, L. (2015). *The onlife manifesto: Being human in a hyperconnected era*. Springer Nature.
- Franco, J. (7 de septiembre de 2010). BaseApp, o como lidiar de mejor manera con Basecamp, hipertextual, <https://bit.ly/3FbnKdy>
- Finogeev, A., Gamidullaeva, L., Bershadsky, A., Fionova, L., Deev, M. & Finogeev, A. (2020). Convergent approach to synthesis of the information learning environment for higher education. *Education and Information Technologies*, 25(1), 11-30. doi:10.1007/s10639-019-09903-5
- From, J. (2017). Pedagogical digital competence: Between values, knowledge and skills. *Higher Education Studies*, 7(2), 43–50. doi: 10.5539/hes.v7n2p43
- García-Peñalvo, F. (2022). Desarrollo de estados de la cuestión robustos: revisiones sistemáticas de literatura. *Education in the Knowledge Society (EKS)*, 23(9)43-64. <https://doi.org/10.14201/eks.28600>
- García-Samaniego, J., Gómez, A., Benítez, J., y Lozada, T. (2022). COVID-19 Cases—Deaths: First approach to the ecuadorian instance. *Tecnologías de la información y la comunicación para estrategias competitivas (ICTCS 2020)*, 191(33)71-76 doi:10.1007/978-981-16-0739-4_7
- García, V., & Fabila, A. (2011). Modelos pedagógicos y teorías del aprendizaje en la educación a distancia. *Apertura*, 3(2), 120-131. <https://www.redalyc.org/articulo.oa?id=68822737011>
- Graesser, A. (08- 09 de junio de 2007). *Emotions and learning with auto tutor* [Conference]. 308 (45), 207-227, Frontiers in artificial intelligence and applications, Inglaterra Cambridge University Press, https://link.springer.com/chapter/10.1007/978-3-642-14363-2_10
- Gilani, M., Inibhunu, C. & Mahmoud, Q. (16-17 de octubre de 2015). *Application and network performance of Amazon elastic compute cloud instances*. [Conference]. Conferencia:

- 4.^a Conferencia internacional IEEE sobre redes en la nube (CloudNet), Chicago.doi:10.1109/cloudnet.2015.7335328
- Giraffa, L., Morales, M. & Uden, M. (2013). Teaching Object-Oriented Programming in First-Year Undergraduate Courses Supported by Virtual Classrooms. En S. P. (SPCOM) (Ed.), *The 2nd international workshop on learning Technology for Education in Cloud*, 8(25), 15–26. doi: 10.1007/978-94-007-7308-0_2
- Gimeno-Sacristán J. & Pérez-Gómez, Á. (1992). *Comprender y transformar la enseñanza*. Morata.
- Giró-Gracia, X., & Sancho-Gil, J. (2022). La Inteligencia Artificial en la educación: Big data, cajas negras y solucionismo tecnológico. *Revista Latinoamericana De Tecnología Educativa - RELATEC*, 21(1), 129-145. <https://doi.org/10.17398/1695-288X.21.1.129>
- González-Martínez, A., Bote-Lorenzo, M., Gómez-Sánchez, E. & Cano, R. (2015). Cloud computing and education: A state-of-the-art survey. *Computers Education*, 80(23), 131-151. doi: <https://doi.org/10.1016/j.compedu.2014.08.017>
- Goleman, D. (1995). *Inteligencia Emocional*, (2da. ed.) Kairós
- Group, Y. (2 de mayo de 2020). Yellow Circle. Obtention de Yellow Circle - free cloud computer lab to learn and teach.: <https://yellowcircle.net/>
- Gumuslouglu, L., y Ilse, A. (abril de 2009). Transformation leadership creativity, and organizational innovation. *Journal of Business Research*, 62(4), 461-473. doi: 10.1108/JSMA-07-2018-0075
- Hayes, B. (2018). Cloud computing. *Communications of the ACM*, 51(7)9-11. doi:10.1145/1364782.1364786
- Hevner, A. (2007). A three-cycle view of design science research. *Scandinavia Journal of Information Systems*, 19(2), 87-92. <https://aisel.aisnet.org/sjis/vol19/iss2/4>
- Hernández-Sánchez, A., y Ortega-Maldonado, A. (2015). Hacia la personalización del e-Learning: la afectividad y su repercusión en el bienestar subjetivo. *Lasallista de Investigación*, 12(2), 194–203. <https://doi.org/10.22507/rli.v12n2a20>

- Hewitt, C. (2008). ORGs for scalable, robust, privacy-friendly client cloud computing. *IEEE Internet Computing*, 12(5), 96-99. doi: 10.1109/MIC.2008.107
- Hevner, A. (2007). A three-cycle view of design science research. *Scandinavian Journal of Information Systems*, 19(2), 87-92. <https://aisel.aisnet.org/sjis/vol19/iss2/4>
- Hevner, A., Park., J. & Ram, S. (2003). Design Science in information systems, Research. *MIS Quarterly*, 28(1), 75– 105. doi:10.5555/2017212.2017217
- Hwang, S. (2007). Utilizing qualitative data analysis software: A review of ATLAS.ti. *Social Science Computer Review*, 26 (4), 519-527. <https://doi.org/10.1177/0894439307312485>
- Hillon-Salgado, D., Gómez-Fierro, W. & Molina-Mosquera, J. (2022). Documentary review about computer programming for the development of computer thinking, logical-mathematical skills, and scientific skills. *Academic Journal of Interdisciplinary Studies*, 11(3), 20-29. doi:10.36941/AJIS-2022-0089
- Horrigan, J. (26 de enero de 2008). *Pew internet and American life project*. Pew Research Center: <https://www.pewresearch.org/topic/internet-technology/>
- Hofmann, P. & Woods, D. (noviembre-diciembre, 2010). Cloud Computing: The limits of public clouds for business applications. *IEEE Internet Computing*, 14(6), 90-93. doi: 10.1109/MIC.2010.136.
- Hocutt, D. & Brown, M. (2018). Glocalizing the composition classroom with google apps for education. Thinking globally, composing locally: Rethinking online writing in the age of the global internet, 5(8), 320-339. doi:10.7330/9781607326649.c015
- Hurwitz, J. Bloor, R. Kaufman, M. & Halper, F. (2010). *Cloud services for dummies*. Wiley Publishing.
- Janev, V., Duduković, J., Jovanović, J. & Vraneš, S. (2009). Web4WeB e-collaboration tools. *Computer Science and Information Systems*, 6(1), 141-168. doi:10.2298/CSIS0901141J

- Jiang, B., Tang, Z., Xiao, X., Yao, J., Cao, R. & Li, K. (2022). Efficient and automated deployment architecture for OpenStack in TianHe Super Computing environment. *IEEE Transactions on parallel and distributed systems*, 33(8), 1811-1824. doi:10.1109/TPDS.2021.3127128
- Jiang, Y., Liu, W., Shi, X. & Qiang, W. (2021). Optimizing the copy-on-write mechanism of docker by dynamic prefetching. *Tsinghua Science and Technology*, 26(3), 266-274. doi:10.26599/TST.2019.9010034
- Kats, Y. (2010). Learning management system technologies and software solutions for online teaching: Tools and applications. *IGI global*, 15(4). doi: [10.4018/978-1-61520-853-1](https://doi.org/10.4018/978-1-61520-853-1)
- Khalidi, Y. (Marzo de 2011). Building a Cloud computing platform for new possibilities. *en Computer IEEE*, 44(3), 29 -34 doi: 10.1109/MC.2011.63.
- Kitchenham, B. & Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering, *Technical Report, Keele University*, 2(1), 6-23. <https://doi.org/10.1145/2372233.2372235>
- Kitchenham, B., Pfleeger, S., Pickard, L., Jones, P., Hoaglin, D., El Emam, K. & Rosenberg, J. (2002). Preliminary guidelines for empirical research in software engineering. *IEEE Transactions on Software Engineering*, 28(8), 721–73. doi: 10.1109/TSE.2002.1027796
- Klymash, M., Tchaikovskyi, I., Hordiichuk-Bublivska, O. & Pyrih, Y. (18-19 de octubre de 2021). *Research of microservices features in information systems using spring boot [Conference]*. EE 2020 International Conference on Infocommunication Problems: Science and Technology (PIC S&T). 507-510. doi:10.1109/PICST51311.2020.9467911
- Ko, A., DeLine, R. & Venolia, G. (2007). *Information needs in collocated software development teams [Conference]*. International conference on software engineering, 344-353. doi:10.1109/ICSE.2007.45

- Kop, R. (2011). Cloud computing and creativity: Learning on a massive open online course. *European Journal of Open, Distance and E-Learning*.12-22.doi: 10.1629/UKSG.577
- Kort, B., Reilly, R. & Picard, R. (2001b). An affective model of interplay between emotions and learning: Reengineering educational pedagogy – building a learning companion. *En memorias de The IEEE International Conference on Advanced Learning Technologies*,6(8),43-46. doi: 10.1109/ICALT.2001.943850.
- Kurysheva, O. (14 de septiembre de 2014) *¿Qué es Alfresco?* Incentro, <http://www.alfresco.com/es/recorrido-por-alfresco>
- Kwon, S. & Le, J. (2020). DIVDS: Docker image vulnerability diagnostic system. *IEEE Access*, 8 (8), 42666-42673.doi: 10.1109/ACCESS.2020.2976874
- Lai, J., Bower, M., De Nobile, J. & Breyer, Y. (2022). What should we evaluate when we use technology in education? *Journal of Computer Assisted Learning*, 38(3), 743– 757. <https://doi.org/10.1111/jcal.12645>
- Lagos, G. (2018). Las plataformas digitales, un nuevo estilo de aprendizaje en la educación superior del Ecuador. *Revista científica SINAPSIS*, 2(13), 458-501.doi: 10.37117/s.v2i13.150
- Lambert, M., Kalyanam, R., Kooper, R. & Yang, B. (2021). Securing CHEESEHub: A cloud-based, containerized cybersecurity education platform. *Conference Proceeding Series*. 1(4),212-219.doi:10.1145/3437359.3465584
- León, A. (2007). Qué es la educación. *Educere*,11(39),595-604. <https://www.redalyc.org/articulo.oa?id=35603903>
- Lemos, A., y Cunha, P. (2003). *Olahares sobre a cibercultura*. Sulina.
- Lewin, D.& Lundie, D. (2016). Philosophies of digital pedagogy. *Studies in philosophy and education*, 35(3), 235–240. doi:10.1007/s11217-016-9514-7
- Lixiang, L. (2010). Study on the affective computing hierarchy model with personality features. *Education Science*,15(7), 543-546. doi:10.3390/educacion15010065

- Lizunkov, V., Morozova, M., Zakharova, A. & Malushko, E. (2021). Evaluation criteria for education institutions-industry collaboration: territories of advanced development (TAD) environment. *Vestnik of Minin University*, 9(1), 1. doi: 10.26795/2307-1281-2021-9-1-1
- Lorenzo, I. (22 de marzo de 2016). 9 usos sorprendentes de la tecnología de reconocimiento facial. Mediatrends. <https://www.mediatrends.es/a/91412/software-reconocimiento-facial/>
- López-Aguilar, D., Álvarez-Pérez, P. y Ravelo-González, Y. (2022). Capacidad de adaptabilidad e intención de abandono académico en estudiantes universitarios. *Revista de Investigación Educativa*, 40(1), 237-255. doi:10.6018/rie.463811
- Lu, T., Hoyer, S., Wang, Q., Hu, L. & Chen, Y. (2021). Distributed data processing for large-scale simulations on cloud. *Joint IEEE International Symposium on Electromagnetic Compatibility Signal and Power Integrity*, 53(1), 53-58. doi:10.1109/EMC/SI/PI/EMCEurope52599.2021.9559316
- Lucas, M., Dorotea, N., y Piedade, J. (2021). Developing teachers' digital competence: Results from a pilot in Portugal. *Revista Iberoamericana de Tecnologías Del Aprendizaje*, 16(1), 84-92. doi:10.1109/RITA.2021.3052654
- Hevner, A., Park, J. & Ram, S. (2003). Design Science in information systems research. *Management information systems quarterly*, 28(1)75–105. <https://doi.org/10.2307/25148625>
- Ionescu, C., Paschia, L., Gudanescu, N., Stanescu, S., Neacsu, V., Coman, M., & Uzlau, M. (2020). Analysis of the E-Learning education system during pandemic period—COVID-19 in Romania. *Sustainability*, 12(21), doi: <https://doi.org/10.3390/su12219030>
- Iglesias, A., Olmos, M., Torrecilla, E. & Mena, J.(2015).Evaluar para optimizar el uso de la plataforma Moodle en el departamento de didáctica, organización y métodos de investigación. *Tendencias Pedagógicas*, 23(6), 155-170. <https://revistas.uam.es/tendenciaspedagogicas/article/view/2077>

- Ilaltdinova, E., Yu. I., Belyaeva, T. & Lebedeva, I. (2019). Pedagogía digital: características del término evolución en el marco de categorías y conceptos de la pedagogía. *Perspectivas de la ciencia y la educación*, 4(40), 33-43 <https://revistas.usal.es/tres/index.php/1130-3743/article/download/31721/29712?inline=1>
- Jornet, J., y Sancho, C. (2018). El valor social de la educación: a modo de introducción. *Revista Fuentes*, 20(2), 11–13.doi: 10.12795/revistafuentes.2018.v20.i2.00
- Kiili, K. & Ketamo, H. (2018). Evaluating cognitive and affective outcomes of a digital game-based math test. *IEEE Transactions on Learning Technologies*, 11(2), 255-263. doi:10.1109/TLT.2017.2687458
- Kitchenham, B. & Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering, *Technical Report, Keele University*, 2(1), 1-2.doi: <https://doi.org/10.1145/2372233.2372235>
- Koh, J. (2020). Three approaches for supporting faculty technological pedagogical content knowledge (TPACK) creation through instructional consultation. *Educ. Technol*, 51(6), 2529-2543. doi: <https://doi.org/10.1111/bjet.12930>
- Kölling, M. & Rosenberg, J. (2001). Guidelines for teaching object orientation with Java. *In Proceedings of the 6th Conference on information technology in computer science education (Tics). Canterbury: IEEE*.33-36. doi: [10.1145/507758.377461](https://doi.org/10.1145/507758.377461)
- Narro, J., Martuscelli, J. y Barzana, E. (2012). *Plan de diez años para desarrollar el Sistema Educativo Nacional*. Dirección General de Publicaciones y Fomento Editorial, UNAM <http://www.planeducativonacional.unam.mx>
- Ma, H. & Yarosh, S. (2021). A review of affective computing research based on function-component-representation framework. in *IEEE Transactions on Affective Computing*, 14(2),1655-1674. doi: 10.1109/TAFFC.2021.3104512.
- Macías, E. (2019). Uso de Cloud Computing en la Educación. *Revista Antioqueña de las Ciencias Computacionales y la Ingeniería de Software*, 9(2), 5.doi: <http://doi.org/10.5281/zenodo.3591998>

- Mandal, S., Salify, S., Ghosh, A., Maji, G., Khatua, S., & Das, R. (2022). An approach to cost minimization with EC2 spot instances using VM based migration policy. *Computer Science*, 13145, 96-110. doi:10.1007/978-3-030-94876-4_6
- Mahesh, G., & Jayamma, K. (2021). *Preferences and experiences of google apps in collaborative teaching and learning*. Library Philosophy and Practice
- Mahmud, E., Rozi, F., Hefniy, Baharun, H., Syakroni, A., Muali, C. & Abdullah, D. (2021). The effect of using edmodo application on students' mastery skill of technology. *Journal of Physics: Conference Series*, 1899, doi:10.1088/1742-6596/1899/1/012157
- Mateus, J., Andrada, P., González, C., Ugalde, C. & Novomisky, S. (2022). Teachers' perspectives for a critical agenda in media education post COVID-19. A comparative study in Latin América. *Comunicar*, 30(70), 9-18. doi:10.3916/C70-2022-01
- Martínez, F. (2018). La preocupación por la calidad de la educación y su valor social. *Revista Fuentes*, 20(2), 17-27. doi: <http://dx.doi.org/10.12795/revistafuentes.2018.v20.i2.01>
- Martín-Conde, M., del Cacho, E., Calvo, E., Roura, J., Pérez, M. & Soy, D. (2021). Improvement of outpatient pharmacy through patient participation and lean methodology. *Farmacia Hospitalaria*, 45(6), 317-322. doi:10.7399/fh.11681
- Medeiros, R., Ramalho, G. & Falcao, T. (2019). A systematic literature review on teaching and learning introductory programming in higher education. *IEEE Transactions on Education*, 62(2), 77-90. <https://eric.ed.gov/?id=EJ1214650>
- Menezes, G., Cafeo, B. & Hora, A. (2022). How are framework code samples maintained and used by developers? the case of android and spring boot. *Journal of Systems and Software*, 185. doi: 10.1016/j.jss.2021.111146
- Mejía, O. & Ballesteros, J. (2014). Computación en la Nube, Tendencia de Importancia y Trascendencia en la Educación Superior. *Ingenio Magno*, 5, 128-136. <http://hdl.handle.net/11634/5002>

- Mell, P. & Grance, T. (2009). The NIST definition of cloud Computing, information technology Laboratory National Institute of Standards and Technology (NIST), *Scientific Research*, 53-50. <https://dl.acm.org/doi/pdf/10.5555/2600239>
- Merkel, D. (2014). Docker: Lightweight linux containers for consistent development and deployment. *Linux Journal*, 239(2), <https://dl.acm.org/toc/linux/2014/2014/239>
- Merino, R. & Lizandra, J. (2022). The hybridization of the pedagogical models of cooperative learning and adventure education as a didactic strategy for enhancing the coexistence and conflict management in class: A practical experience from the physical education and tutoring lessons. *Retos*, (43), 1037-1048. doi:10.47197/RETOS.V43I0.86289
- Mitchell, T., Betteridge, J., Carlson, A., Hruschka, E. & Wang, R. (2009). Populating the semantic web by macro-reading internet text, en *International Semantic Web Conference*, 5823,998-1002. Doi:10.1007/978-3-642-04930-9_66
- Microsoft. (22 de mayo de 2020). *¿Qué es la informática en la nube?* Microsoft, <https://bit.ly/3MCvTui>
- Microsoft. (2020). *Microsoft Azure. ¿Qué es Azure?* Microsoft, <https://bit.ly/3RSYwa6>
- Mohanty, S., Potluri, S., Prakash, V., Srinath, B. & Manjunath, B. (2021). Cloud security concepts, threats and solutions: Artificial intelligence-based approach. *Cloud security: Techniques and applications*, 1-20. doi:10.1515/9783110732573-001
- Moghim, M., Stone, R. & Rotshtein, P. (2020). Affective recognition in dynamic and interactive virtual environments. *IEEE Transactions on Affective Computing*, 11(1), 45-62. doi:10.1109/TAFFC.2017.2764896
- Montoya, L., Parra, M., Lescay, M., Cabello, O. & Coloma, G. (2019). Teorías pedagógicas que sustentan el aprendizaje con el uso de las Tecnologías de la Información y las Comunicaciones. *Revista Información Científica*, 98 (2), 241-255. <https://www.redalyc.org/articulo.oa?id=551760346011>

- Moro-Cabero, M., & Llanes-Padrón, D. (2019). Professional perspectives in cloud computing environments. *Anales de Investigación*, Vol. 15(1), 11-34.
10.13140/RG.2.2.26008.88329
- Muñoz, J., y Sahagún, M. (2011). Análisis cualitativos asistido por ordenador con ATLAS.ti. *Nuevas Perspectivas Conceptuales y Metodológicas*, 5(9),299–363.
<https://www.redalyc.org/articulo.oa?id=86819565014>
- Murphy, B., Rocchi, M. (2021). Ethics and Cloud Computing. In: Lynn, T., Mooney, J.G., van der Werff, L., Fox, G. (eds) *Data Privacy and Trust in Cloud Computing*. Palgrave Studies in Digital Business & Enabling Technologies. Palgrave Macmillan, Cham.
https://doi.org/10.1007/978-3-030-54660-1_6
- Omer, U., Farooq, M.& Abid, A. (2020). Cognitive Learning Analytics Using Assessment Data and Concept Map: A Framework-Based Approach for Sustainability of Programming Courses. *Sustainability*,12(17).doi: <https://doi.org/10.3390/su12176990>
- Organización para la Cooperación y el Desarrollo Económicos (OCDE). (3 march 2012). *Education at a Glance: OCDE Indicators*. <https://bit.ly/3kEJTrf>
- Ortony, A., Clore, G. L., y A. Collins. (10 june 1988). *The Cognitive Structure of Emotions*, Cambridge University Press. <http://dx.doi.org/10.1017/CBO9780511571299>
- Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura. (2008). *Reflexiones en torno a la evaluación de la calidad educativa en América Latina y el Caribe*. Laboratorio latinoamericano de evaluación de la calidad en educación (Llece).
<http://unesdoc.unesco.org/images/0017/001776/1776>
- Olugbenga, W., Ewuzie, I., Usoro, A. & Cannolly, T. (2018). E-Learning to m-Learning: framework for data protection and security in cloud infrastructure. *International journal of informatio technology and computer science (IJITCS)*, 10(4), 1-9.doi: <https://doi.org/10.5815/ijitcs.2018.04.01>
- Papert, S. (2020). *Mindstorms: Children, computers, and powerful ideas*. Basic books.

- Paterson, J., Adams, J., White, L., Csizmadia, A., Erdil, D., Foster, D. & Stott, L. (2021). Designing dissemination and validation of a framework for teaching cloud fundamentals. *in Computer Science Education, ITiCSE*, 163-181. doi:10.1145/3502870.3506569
- Pentland, A. & Choudhury, T. (2000). Face recognition for smart environments. *Computer*, 33(2), 50–55. <https://doi.org/10.1109/2.820039>
- Peppers, K. et al. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45–77. <https://doi.org/10.2753/MIS0742-1222240302>
- Pezeshki, C. & Racicot, K. (24 june 2006). *Improving performance and reducing professor workload in a capstone design class using project management software*. [Conference]. Paper presented at the ASEE Annual Conference and Exposition, Conference Proceedings, Chicago, Illinois, doi: 10.18260/1-2--206
- Picard, R. (1997). *Affective Computing* United Stated: MIT Press. <https://mitpress.mit.edu/9780262661157/affective-computing/>
- Picard, R. (2000 a). *Affective Computing*, MIT Press
- Picard, R., & Klein, J. (2002 b). Computers that recognise and respond to user emotion: theoretical and practical implications. *Interacting with Computers*, 14(2), 141–169. doi:10.1016/S0953-5438(01)00055-8
- Piaget, J. (2005). *Inteligencia y afectividad*, Aique
- Polze, A. (2009). *A comparative Analysis of Cloud Computing Environments*. University of Potsdam.
- portalSaS.com. (4 de enero de 2012). *Gestión de Proyectos*, portalSaS.com. <http://www.portalsaas.com/Basecamp.html>
- Plowman, L. & McPake, J. (2013). Seven myths about young children and technology. *Childhood Education* ,89(1), 27–33. <https://doi.org/10.1080/00094056.2013.757490>

- Puviani, L., Rama, S. & Vitetta, G. (2012) A mathematical description of emotional processes and its potential applications to affective computing," in *IEEE Transactions on Affective Computing*,12(3),692-706, doi: 10.1109/TAFFC.2018.2887385.
- profesorPaco. (2011). *¿Qué es Edmodo?* profesorPaco. <https://bit.ly/3s6prE1>
- Quezada-Sarmiento, P., Chango-Cañaveral, P. , Fraiz-Brea, J., Armijos-Lancha, K. & Ponce, M.(19-21 june 2021). *Tourism and gastronomy management through an educational web platform on Yacuambi of the province of Zamora Chinchipe Ecuador*. [Conference], Paper presented at the WCSE 2020: 2020 10th International Workshop on Computer Science and Engineering, España, 233-238. doi:10.18178/wcse.2020.06.036
- Quezada, P., Enciso, L. & Garbajosa, J. (04-06 june 2015). *Using tools of cloud computing for defining, planning, monitoring innovation projects and knowledge generation*. [Conference] Paper presented at the XI Jornadas Iberoamericanas de Ingeniería De Software e Ingeniería del Conocimiento, JIISIC, Portugal, 121-131, <https://www.risti.xyz/issues/ristie15.pdf>
- Quezada-Sarmiento, P. & Suárez–Guerrero, C. (2021). Cloud computing in the training process in web programming. *RISTI - Revista Ibérica de Sistemas e Tecnologias De Informacao*, 42,10-19. <https://www.risti.xyz/issues/ristie42.pdf>
- Quezada, P., Enciso, L., Mengual, S. & Hernández, W. (2018). *Integration of cloud computing tools and knowledge bodies for the management of programming projects*. [Conference],13th Iberian Conference on Information Systems and Technologies, CISTI ,1-6 Caceres; Spain: IEEE Computer Society. [10.23919/CISTI.2018.8399219](https://doi.org/10.23919/CISTI.2018.8399219)
- Quezada-Sarmiento, P. & Mengual-Andrés, S. (2019). *Cloud computing and affective computer approach on development programming skills*. [Conference], Paper presented at the Iberian Conference on Information Systems and Technologies, CISTI, Portugal, 10-19. doi:10.23919/CISTI.2019.8760697

- Quezada, P., Jumbo, L. & Hernández, W. (2020). *Knowledge representation model for bodies of knowledge based on design patterns and hierarchical graphs*, in. *Computing in Science y Engineering*, 55-63. doi: 10.1109/MCSE.2018.287
- Quezada-Sarmiento, P., Suárez-Guerrero, C., Narváez-Ríos, M. & Gutiérrez-Albán, L. (2022). Use of cloud computing tools on pedagogical and educational contexts. *Lecture notes in networks and systems*, 468, doi:10.1007/978-3-031-04826-5_35
- Rahim, R., Masitoh, F., Kuzmichev, A., Lydia, E. & Shankar, K. (2019). Green cloud computing ideas with security threats and solutions. Ideas de computación en la nube verde con amenazas y soluciones de seguridad. *Solutions. Opción* 35(20),1101-1121, <https://produccioncientificaluz.org/index.php/opcion/article/view/30949>
- Ramos, A. (1 de mayo de 2020). *Redmine, software libre de gestión de proyectos*. Incubaweb. <https://bit.ly/3y7APDg>
- Red Hat, Inc. (2020). *¿Qué es DOCKER?* Red Hat, Inc. <https://www.redhat.com/es/topics/containers/what-is-docker>
- Resende, L. (2017). La educación en el laberinto tecnológico: de la escritura a las máquinas digitales. *Revista Científica*, 43. 02-206. doi: <https://doi.org/10.5585/eccos.n43.7592>
- Robinson, D. (2008). Amazon web services made simple: learn how amazon ec2, s3, simpledb and sqs web services enables you to reach business goals faster. *Emereo Pty Ltd, London, GBR*. <https://www.sciepub.com/reference/398243>
- Rodríguez-Arboleda, V., Gallar-Pérez, Y., y Barrios-Queipo, E. (2017). Consideraciones teóricas acerca de la Computación Afectiva en el proceso de enseñanza aprendizaje de la Educación Superior. *CienciAmérica*, 6(3), 170-175. <https://bit.ly/3yN2IEP>
- Roman, F. (2018). *Microsoft Azure – The Cloud Services to Meet the Increasing Business's Needs*. <https://bit.ly/3OLi9iC>
- Romero, L., Santucci, V., Gentile, C., Sklar, D. & Ale, M. (2020). *Competency-based learning objects: A methodology for their development in engineering careers [Conference]*.

- Paper presented at the 2020, Biennial Congress of Argentina, Argencon, doi:10.1109/ARGENCON49523.2020.9505399
- Russo, J. y Ellabidy M. (Diciembre, 2013). *Using Amazon web services to teach web-enabled database and networking concepts. Comput. Sci.*
- Salcedo-Lagos, P., Morales-Candia, S., Fuentes-Riffo, K., Rivera-Robles, S., & Sanhueza-campos, C. (2021). Teachers' perceptions analysis on students' emotions in virtual classes during covid19 pandemic: A lexical availability approach. *Sustainability (Switzerland)*, 13(11), 234-251.doi:10.3390/su13116413
- Salcedo, H. (2011). Los objetivos y su importancia para el proceso de enseñanza-aprendizaje. *Revista de Pedagogía*, XXXII, (91),113-130.<https://www.redalyc.org/articulo.oa?id=65926549007>
- Sangrà, A., Badia A., Cabrera, N., Lanzo A., Espasa, M., Fernández, L., Guàrdia, T., Guasch P., Montse, G., Maina, M., Raffaghelli, J., Romero, M. & Romeu, T. (2020). *Decálogo para la mejora de la docencia online Published, Editorial. UOC* <http://hdl.handle.net/10609/122307>
- Sclater, N. (2010). *Cloud computing in education*. instituto de la unesco para la utilización de las tecnologías de la información en la educación. <https://unesdoc.unesco.org/ark:/48223/pf0000190432>
- Sánchez, T. (2021). Implementación de rediseños y diseños de carreras para el mejoramiento de la calidad de la Educación Superior del Ecuador [Tesis doctoral inédita]Universidad Multiversidad Mundo Real Edgar Morin México.
- Severin, E. (2013). *Enfoque estratégico sobre las TICS en la Educación en América Latina y el Caribe. Santiago: Oficina Regional de Educación para América Latina y el Caribe. UNESCO*
- Sein, M. K., et al. (2011). *Action design research. MIS Quarterly*, 35(1), 37–56. <https://misq.umn.edu/action-design-research.html>

- Social Media. (9 de febrero de 2014). Social Media. ¿Qué es Redbooth?: Plataforma de colaboración y comunicación: <https://bit.ly/3saVq5V>
- Soni, P. K., & Dhurwe, H. (2024). Challenges and open issues in cloud computing services. In *Advanced computing techniques for optimization in cloud* (1st ed., pp. 19). Chapman and Hall/CRC. <https://doi.org/10.1201/9781003457152>
- Sousa, P., Faria, P., Correia, M., Resende, J. & Antunes, L. (2016). *Digital signatures workflows in alfresco*, 9831, doi:10.1007/978-3-319-44159-7_22
- Sosa, R., García S, Sánchez A, Moreno A. & Reinoso A.(2015). B-Learning y teoría del aprendizaje constructivista en las disciplinas informáticas: un esquema de ejemplo a aplicar. *Recent res dev lear technol*. http://1.asset.soup.io/asset/2112/4657_4fa5.pdf
- Suárez-Guerrero, C., Ros-Garrido, A. & Lizandra, J. (2021). Approach to digital teaching competence in vocational training. [Aproximación a la competencia digital docente en la formación profesional]. *Revista de Educación a Distancia*, 21(67). doi:10.6018/RED.431821
- Suárez-Guerrero, C., Gutiérrez-Esteban, P. & Ayuso-Delpuerto, D. (2024). Pedagogía Digital. Revisión sistemática del concepto . Teoría De La Educación. *Revista Interuniversitaria*, 36(2), 157–178. <https://doi.org/10.14201/teri.31721>
- Slama, J., Niculcea, A., Cancho, M., Jiménez, M. Ibarra, I., López, E. Corsini, J., Gregsamer, C. (2010). *Cloud Computing, la tercera ola de la tecnología de la información*. Fundación de la Innovación Bankinter.
- Solomon D. & Kamalakkannan, S. (2022). Hybrid optimization-based privacy preservation of database publishing in cloud environment. *Concurrency and Computation: Practice and Experience*., 934-946, doi:10.1002/cpe.6844
- Sawyer, R. (2008). *Cambridge handbook of the learning Sciences*. (Cambridge, Ed.) London: Cambridge University Press.
- Sclater, N. (2010). eLearning in the cloud. *international journal of virtual and personal, Learning Environments (IJVPLE)* 10-19. doi:10.4018/jvple.2010091702(1), 10-19.

- Shi, T., Ma, H., Chen, G., & Hartmann, S. (2022). Cost-effective web application replication and deployment in multi-cloud environment. *IEEE Transactions on Parallel and Distributed Systems*, 33(8), 1982-1995. doi:10.1109/TPDS.2021.3133884
- Shu, R., Gu, X. & Enck, W. (2017). A study of security vulnerabilities on docker hub. Association for Computing Machinery, Nueva York
- SocialMedia. (9 de febrero de 2014). Social Media. ¿Qué es Redbooth?: Plataforma de colaboración y comunicación, 269-280. doi: <https://bit.ly/3saVq5V>
- Sousa, P., Faria, P., Correia, M., Resende, J., y Antunes, L. (2016). Digital signatures workflows in alfresco. *Lecture notes in computer Science*, 9831. doi:10.1007/978-3-319-44159-7_22
- Sultan, N. (2010). Cloud computing for education: Anew daw?. *International Journal of Information Management*, 30(2), 109-116. <http://dx.doi.org/10.1016/j.ijinfomgt.2009.09.004>
- Stratabi. (26 de mayo de 2020). *Alfresco gestión documental Tecnologías Alfresco*, Stratabi. <https://www.stratabi.com/alfresco>
- Taurion, C. (2009). Cloud Computing: Transformando o mundo da tecnologia da informação. BRASPORT.
- Tao, J.& Tan, T. (2005). Affective computing: a review. Affective computing and intelligent interaction, *Lecture Notes in Computer Science*, 3784, 981–995. https://doi.org/10.1007/11573548_125
- Tan, X.& Kim, Y. (2011). Cloud computing for education: A case of using google docs in MBA group projects. Paper presented at the *Proceedings of the 2011 International Conference on Business Computing and Global Informatization, BCGIn 2011*, 641-644. doi:10.1109/BCGIn.2011.169
- Tarigan, J., Zamzami, E., Jaya, I., Hardi, S. & Zarlis, M. (2019). Ability level of git hub amongst computer science students. *International Journal of Recent Technology and Engineering*, 8(2), 300-303. doi:10.35940/ijrte. B1028.0782S719

- Tagra, A., Zhang, H., Rajbahadur, G., & Hassan, A. (2022). Revisiting reopened bugs in open-source software systems. *Empirical Software Engineering*, 27(4). doi:10.1007/s10664-022-10133-6
- Tavares, D., De Menezes, O. & De Nevado, R. (2012). Pedagogical architectures to support the process of teaching and learning of computer programming. En A. E. IEEE Education Society (Ed.), *proceedings - frontiers in education conference, fie. seattle, wa*; United States: IEEE. <https://doi.org/10.1007/s10664-022-10133-6>
- Tian, L., Oviatt, S., Muszynski, M., Chamberlain, B., Healey, J. & Sano, A. (2022). Affective Computing for Enhancing Well-Being. *Applied Affective Computing. Association for Computing Machinery, New York*. <https://doi.org/10.1145/3502398.3502408>
- Torres, M., Zea, M., Pérez. & Martínez, Y. (2022). Project based learning as a teaching trend in higher education. *Universidad y Sociedad*.14(1), 53-58. doi: <https://doi.org/10.3390/educsci14010053>
- Torres, C., y Alcántar, M. (2011). Uso de las redes sociales como estrategias de aprendizaje. ¿Transformación educativa? *Apertura*, 3(2), <https://www.redalyc.org/articulo.oa?id=68822737001>
- Tuncay, E. (2010). effective use of cloud computing in educational institutions. *Procedia Social Behavioral Sciences*, 938-942. <https://www.redalyc.org/articulo.oa?id=68822737001>
- Universidad Técnica Particular de Loja. (12 de noviembre de 2021). Universidad Técnica Particular de Loja. (UTPL, Productor) <https://www.utpl.edu.ec/es/acercade>
- Universidad Técnica Particular de Loja. (12 de noviembre de 2021). *Carrera de Computación, Digital*. Carrera de Computación: <https://www.utpl.edu.ec/carreras/computacion>
- Universidad Nacional de Loja (2021). *Computación, 2.0. (UNL, Editor), Computación*: https://www.unl.edu.ec/oferta_academica/facultad-de-la-energia-las-industrias-y-los-recursos-naturales-no-renovables/computacion-2019
- Universidad Internacional del Ecuador (2021) <https://www.uide.edu.ec/quienes-somos/>

- Universidad Internacional del Ecuador (2021), *Programas académicos*, UIDE, <https://www.uide.edu.ec/historia/>
- Universidad Nacional de Loja, (2021). *Escenarios de aprendizaje, investigación y vinculación*, UNL, <https://unl.edu.ec/noticia/unl-cumple-161-anos>
- Valdez, A., Octava, H., Gómez, H., & Vizcaino, A. (2020). Sentiment analysis in Jira software repositories. *Paper presented at the Proceedings - 2020 8th Edition of the International Conference in Software Engineering Research and Innovation, CONISOFT 2020*, 3(1),254-259. doi:10.1109/CONISOFT50191.2020.00043
- Van den Akker, J. y Plomp, T. (1993). Development research in curriculum: Propositions and experiences. *Springer, Dordrecht*. https://doi.org/10.1007/978-94-011-4255-7_1
- Van, J. (2004). Management research based on the paradigm of the design sciences: The quest for field-tested and grounded technological rules. *Journal of Management Studies*, 41(2), 219–246. doi: 10.1111/j.1467-6486.2004. 00430.x
- Van, J. (2005). Management research as a design science: Articulating the research products of mode 2 knowledge production in management. *British Journal of Management*, 16(1), 19–36. <https://doi.org/10.1111/j.1467-8551.2005.00437.x>
- Vaillant, D., y García, M. (2012). Ensinando a ensinar: as quatro etapas de uma aprendizagem. UTFPR
- Wang, F., y Hannafin, M. (2005). Design-Based research and technology- enhanced learning environments. *Educational Technology Research and Development*, 53(4), 5–23. <https://doi.org/10.1007/BF02504682>
- Velásquez, J. (1997). Modeling emotions and other motivations in synthetic agents. *En memorias de The Fourteenth National Conference on Artificial Intelligence*, <https://dl.acm.org/doi/10.5555/1867406.1867408>

- Vygotsky, L. (1994). The problem of the environment.1994. 338- 354. doi:10.15448/1981-2582.2018.3.31795
- Vera, J., y Argüello, B.(2019). El aprendizaje de la programación de computadoras para futuros docentes informáticos. *Revista Iberoamericana de la Educación*, 2(1). <https://doi.org/10.31876/ie.v2i1.12>
- Vidal, I., López, B., y Otero, L. (2021). New digital skills in students empowered with the use of augmented reality. pilot study. *RIED-Revista Iberoamericana de Educación a Distancia*, 24(1), 137-157. doi:10.5944/ried.24.1.27501
- Villapudua, K., Ley, D., y Arteaga, M. (2021). Precariedad digital y el impacto del COVID-19 en la educación superior: Un estudio narrativo en la frontera norte de México. *Journal of Iberian and Latin American Research*, 27(3), 489-505. doi:10.1080/13260219.2021.2030285
- Vorobeva, I., Panov, A., Safronov, A. & Sazonov, A. (2022). Cloud computing models for business. *international journal of emerging technology and advanced engineering*, 12(1), 163-172. doi:10.46338/IJETAE0122_16
- Unal, E. & Uzun, A. (2021). Understanding university students' behavioral intention to use edmodo through the lens of an extended technology acceptance model. *British Journal of Educational Technology*, 52(2), 619-637. doi:10.1111/bjet.13046
- XACATA (2021) *Qué es GitHub y qué es lo que les ofrece a los desarrolladores*. XACATA <https://www.xataka.com/basics/que-github-que-le-ofrece-a-desarrolladores>
- Wei, W. (2022). A new "offline + online" teaching model of college English based on cloud computing. *Forest Chemicals Review*, 255-263. doi:10.17762/jfcr.vi.207
- Wing, J. (2011). Cuaderno de investigación: Pensamiento computacional: qué y por qué. *Revista The Link*, 6, 20-23. doi: <https://people.cs.vt.edu/~kafura/CS6604/Papers/CT-What-And-Why.pdf>

- Wing, J. (2011). Research notebook: Computational thinking—what and why? The link magazine. *Communications of the ACM*, 10.1145/1118178.1118215
- Wu, C., Huang, Y. & Hwang, J. (2015). Review of affective computing in education/learning: Trends and challenges. *British Journal of Educational Technology*, 47(6), 1304–1323, doi:10.1111/bjet.12324.
- Xue, Q., Zhu, Y. & Wang, J. (2022). Mean estimation over numeric data with personalized local differential privacy. *Frontiers of Computer Science*, 16(3) doi:10.1007/s11704-020-0103-0
- Yasar, R. (2018). Mitigating docker security. *Issue*. 1(3). doi: <https://doi.org/10.48550/arXiv.1804.05039>
- Yan, X., Lin, Z., Zhang, X. & Yang, W. (2014). *Design and realization of micro-information system based on alfresco development tool* [Conference]. Paper presented at the Proceedings - 2014 International conference on information science. electronics and electrical engineering, ISEEE, 3, 1604-1608. doi:10.1109/InfoSEEE.2014.6946192
- Zagalsky, A., Feliciano, J., Storey, M., Zhao, Y. & Wang, W. (2015). *The emergence of GitHub as a collaborative platform for education* [Conference]. Paper presented at the CSCW 2015 - Proceedings of the 2015 ACM International Conference on Computer-Supported Cooperative Work and Social Computing, 1906-1917. doi:10.1145/2675133.2675284
- Zhang, F., Sun, G., Zheng, B. & Dong, L. (2021a). Design and implementation of energy management system based on spring boot framework. *Information (Switzerland)*, 12(11), 457-514. doi:10.3390/info12110457
- Zhang, F., Sun, G., Zheng, B. y Dong, L.(2021 b). Diseño e implementación de un sistema de gestión de energía basado en el marco Sprint boot. *Información* , 12 (11), 457. <https://doi.org/10.3390/info12110457>

- Zhang, J., Yang, T. & Li, W. (2022). Optimal locating method of edge computing device in cyber physical distribution system. *Energy Reports*, 8(7), 684-694. doi: 10.1016/j.egyr.2022.01.247
- Zou, J., Feudally, D, Kumar, N., Wang, H. & Choo, K. (2022). Integrated blockchain and cloud computing systems: A systematic survey, solutions, and challenges. *ACM Computing Surveys*, 54(8), 346-362. doi:10.1145/3456628

Anexos

Anexo 1: Ficha de Observación de Clase

Institución Educativa:

Profesor:

Fecha:

Duración de la Clase:

Plataforma de Computación en la Nube Utilizada:

1. Objetivos de la Clase

- Descripción Breve:
 - ¿Cuáles son los objetivos de aprendizaje específicos de la clase?
- Alineación con el Programa:
 - ¿Cómo se alinean estos objetivos con el programa general del curso?

2. Herramientas de Computación en la Nube

- Plataformas y Servicios Utilizados:
 - Enumerar las herramientas de computación en la nube utilizadas (p. ej., AWS, Google Cloud, Azure).
- Integración Tecnológica:
 - ¿Cómo se integran estas herramientas de computación en la nube para la entrega de contenido y la interacción en clase?
- Accesibilidad y Usabilidad:
 - Observar y registrar cualquier dificultad que los estudiantes enfrenten al utilizar estas herramientas.

3. CA

- Herramientas y Técnicas Utilizadas:
 - Describir cualquier software o hardware de CA
 - implementado (p. ej., análisis de expresiones faciales, tono de voz).
- Respuesta Emocional de los Estudiantes:

- Observaciones sobre la respuesta emocional de los estudiantes durante la clase.
- Impacto en la Participación y el Aprendizaje:
 - ¿Cómo influyen las tecnologías de CA en la participación y el aprendizaje de los estudiantes?

4. Interacción y Participación

- Dinámicas de Clase:
 - Descripción de cómo los estudiantes interactúan entre ellos y con el profesor.
- Participación:
 - Nivel de participación de los estudiantes en actividades y discusiones.

5. Evaluación y Feedback

- Métodos de Evaluación Utilizados:
 - ¿Qué métodos de evaluación se utilizan y cómo se aplican en un entorno en la nube?
- Feedback Instantáneo:
 - Observaciones sobre cómo el feedback (automatizado o del profesor) es proporcionado y recibido.

6. Desafíos y Oportunidades

- Desafíos Técnicos y Pedagógicos:
 - Cualquier problema técnico o pedagógico observado durante la clase.
- Oportunidades de Mejora:
 - Sugerencias para mejorar la integración de las tecnologías y la enseñanza.

7. Comentarios Generales

- Impresiones Generales:
 - Comentarios y observaciones finales sobre la sesión.

Instrucciones para el Observador

- Preparación:

- Familiares con las herramientas de CN y CA.
- Durante la Clase:
 - Toma notas detalladas, enfocándote en los puntos de observación listados.
- Post-Clase:
 - Completa la ficha basándote en tus notas y discute tus observaciones con el profesor si es posible.

Anexo 2:Entrevista para Clases de Programación

Información del Entrevistado:

- **Nombre:**
- **Rol: (Profesor/Estudiante)**
- **Fecha:**

1. Experiencia General

- ¿Cómo describirías tu experiencia general con el uso de herramientas de CN y CA en clases de Programación?

2. Herramientas de CN

- ¿Qué plataformas de CN has utilizado en tus clases de Programación?
- ¿Cómo crees que el uso de la CN ha afectado tu aprendizaje o enseñanza en esta área?
- ¿Puedes compartir alguna experiencia donde la CN facilitó o dificultó algún aspecto del aprendizaje?

3. CA

- ¿Has experimentado el uso de tecnologías de CA en el aula? Si es así, ¿podrías describir cómo se utilizaron?
- En tu opinión, ¿cuál ha sido el impacto de la CA en la dinámica de la clase y en el aprendizaje individual?
- ¿Has notado alguna diferencia en la interacción o en la respuesta emocional de los estudiantes (o tuya) debido a la implementación de estas tecnologías?

4. Integración Tecnológica

- Desde tu perspectiva, ¿cómo ha sido la integración de estas tecnologías en el currículo de Programación?
- ¿Hay desafíos que has enfrentado al integrar CN y CA en el aula? ¿Cómo los has superado?

5. Impacto en el Aprendizaje y Participación

- ¿Crees que el uso de estas tecnologías ha mejorado la participación en clase? ¿Puedes dar un ejemplo?
- ¿Cómo ha influenciado la CN y la CA en tu motivación o la de tus estudiantes para aprender Programación Avanzada?

6. Futuro y Mejoras

- ¿Qué mejoras sugerirías para el uso de estas tecnologías en futuras clases de Programación?
- ¿Ves un futuro donde la CN y la CA sean componentes estándar en la educación en programación? ¿Por qué o por qué no?

Instrucciones para el Entrevistador

- Preparación:
 - Asegúrate de entender las herramientas y tecnologías mencionadas por el entrevistado.
- Durante la Entrevista:
 - Crea un ambiente relajado y abierto para que el entrevistado se sienta cómodo compartiendo sus experiencias y opiniones.
- Post-Entrevista:
 - Resume y analiza las respuestas para obtener insights y posibles mejoras en la implementación de la tecnología en clases

Anexo 3: Evaluación a nivel de Código

Analizar el código de programación desarrollado por un estudiante de programación web y móvil requiere un enfoque estructurado para identificar fortalezas, áreas de mejora y oportunidades de aprendizaje. A continuación, se presenta los aspectos evaluados a nivel de programación.

1. Revisión Inicial: Objetivo del Proyecto

- Comprende el propósito del proyecto: ¿Qué funcionalidad o problema busca resolver?
- Requisitos: Verifica si el código cumple con los requisitos del proyecto o tarea asignada.
- Plataforma: Identifica si el código está dirigido a una aplicación web, móvil o ambas.

2. Análisis de Estructura del Código

- Organización de carpetas y archivos:
 - ¿Sigue una estructura lógica? Ejemplo: separa los controladores, vistas, modelos, etc.
 - En móviles, ¿usa buenas prácticas como MVVM, MVP o MVC?
- Nombres de archivos y carpetas:
 - ¿Son claros y descriptivos?
 - ¿Siguen un estándar de nomenclatura (camelCase, PascalCase, snake_case)?

3. Estilo de Código

- Legibilidad:
 - ¿El código es fácil de leer?
 - ¿Utiliza sangrías consistentes y espaciado adecuado?
- Nomenclatura de variables, funciones y clases:
 - ¿Son nombres significativos?
 - ¿Siguen convenciones comunes? Ejemplo: camelCase para variables, PascalCase para clases.
- Comentarios:
 - ¿Existen comentarios útiles para comprender el código?
 - ¿Evita comentarios redundantes o innecesarios?

4. Calidad del Código

- Modularidad:
 - ¿El código está dividido en funciones, métodos o clases reutilizables?
- Evitar redundancias:

- ¿Se repite código que podría ser refactorizado?
- Uso de buenas prácticas:
 - Ejemplo en JavaScript: ¿Se utiliza let/const en lugar de var?
 - Ejemplo en programación móvil: ¿Optimiza los ciclos de vida de las actividades/fragments?

5. Funcionalidad

- Pruebas de uso:
 - ¿El proyecto cumple con las funciones descritas?
 - ¿Detectas errores o fallos durante la ejecución?
- Flujo de usuario:
 - ¿La navegación es fluida y sin bugs?
 - ¿La interfaz es intuitiva?
- Adaptabilidad:
 - En web: ¿Es responsive (se adapta a distintos dispositivos)?
 - En móvil: ¿Es compatible con diferentes resoluciones de pantalla?

6. Seguridad

- Validación de datos:
 - ¿Se valida el input del usuario tanto en frontend como en backend?
- Uso de buenas prácticas de autenticación/autorización:
 - En web: ¿Se manejan bien las sesiones o tokens?
 - En móvil: ¿Se protegen los datos sensibles?

7. Optimización y Desempeño

- Eficiencia del código:
 - ¿Evita consultas o cálculos innecesarios?
- Carga y velocidad:
 - En web: ¿Optimiza el uso de recursos (imágenes, scripts, CSS)?
 - En móvil: ¿Minimiza el uso de memoria y batería?

8. Conformidad con Tecnologías Usadas

- Frameworks y librerías:
 - ¿Se aprovechan las capacidades del framework (React, Angular, Flutter, ¿etc.)?
 - ¿Se evitan dependencias innecesarias?
- Estándares del lenguaje:
 - ¿Se sigue la versión recomendada? (Ejemplo: ES6+ para JavaScript)

9. Pruebas Automatizadas

- Cobertura de pruebas:
 - ¿Incluye pruebas unitarias, funcionales o de integración?

- Calidad de las pruebas:
 - ¿Evalúan casos reales y extremos?

10. Retroalimentación

- Identifica logros:
 - Menciona las fortalezas del código, como su creatividad, funcionalidad o limpieza.
- Ofrece sugerencias:
 - Enfócate en áreas de mejora específicas con ejemplos concretos.
- Motivación:
 - Refuerza el aprendizaje al destacar los avances y ofrecer recursos adicionales (tutoriales, guías)