



Universidad Tecnológica ECOTEC

Facultad de Ingenierías

Título del trabajo:

Evaluación de la efectividad del software asistido por IA para la supervisión de los actos de deshonestidad académica en las evaluaciones teóricas de entornos virtuales en instituciones de educación superior de Ecuador.

Línea de Investigación:

Tecnologías de la Información y Comunicación

Modalidad de titulación:

Propuesta Tecnológica

Carrera:

Ingeniería de Software

Título para obtener:

Ingeniero de Software

Autor (a):

César Eduardo Duque Romero

Tutor (a):

Luis Enrique Espinoza Mendoza

Samborondón – Ecuador

2023

**UNIDAD DE INTEGRACIÓN CURRICULAR
CERTIFICADO DE APROBACIÓN DEL TUTOR METODOLÓGICO Y CIENTÍFICO PARA LA
PRESENTACIÓN DEL TRABAJO DE INTEGRACIÓN CURRICULAR**

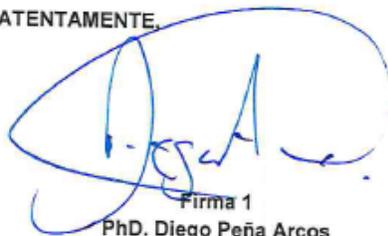
Samborondón, 6 de diciembre de 2023

Magíster
Erika Ascencio Jordán
Decana de la Facultad
Ingenierías
Universidad Tecnológica ECOTEC

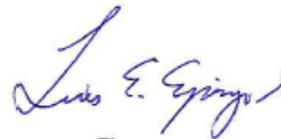
De mis consideraciones:

Por medio de la presente comunico a usted que el trabajo de integración curricular TITULADO: **"Evaluación de la efectividad del software asistido por IA para la supervisión de los actos de deshonestidad académica en las evaluaciones teóricas de entornos virtuales en instituciones de educación superior de Ecuador."** según su modalidad PROYECTO DE INTEGRACIÓN CURRICULAR; fue revisado, siendo su contenido original en su totalidad, así como el cumplimiento de los requerimientos establecidos en la guía para su elaboración, Por lo que se autoriza al estudiante: **DUQUE ROMERO CESAR EDUARDO**, para que proceda con la presentación oral del mismo.

ATENTAMENTE,



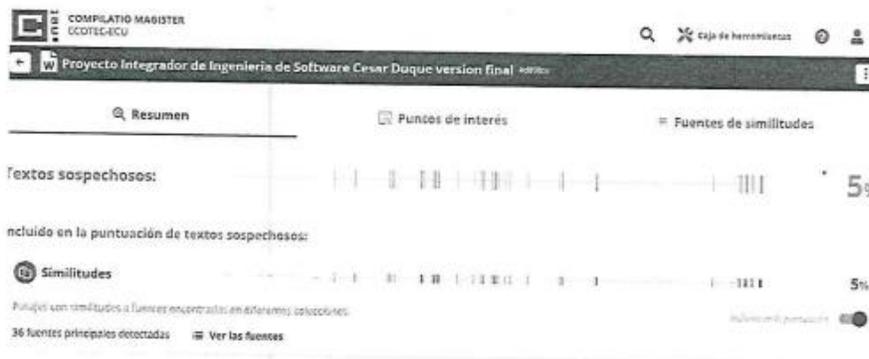
Firma 1
PhD. Diego Peña Arcos
Tutor(a) metodológico



Firma 2
Mgtr. Luis Espinoza
Tutor(a) de la ciencia

**UNIDAD DE INTEGRACIÓN CURRICULAR
CERTIFICADO DEL PORCENTAJE DE COINCIDENCIAS
DEL TRABAJO DE INTEGRACIÓN CURRICULAR**

Habiendo sido revisado el trabajo de integración curricular TITULADO: *Evaluación de la efectividad del software asistido por IA para la supervisión de los actos de deshonestidad académica en las evaluaciones teóricas de entornos virtuales en instituciones de educación superior de Ecuador*, según su modalidad PROYECTO DE INTEGRACIÓN CURRICULAR elaborado por César Eduardo Duque Romero, fue remitido al sistema de coincidencias en todo su contenido el mismo que presentó un porcentaje de coincidencias del (%) 5 mismo que cumple con el valor aceptado para su presentación que es inferior o igual al 10% sobre el total de hojas del Trabajo de integración curricular. Se puede verificar el informe en el siguiente link: <https://app.compileio.net/v5/report/48a08bfacdf61d2358754276505a019343b7c647/su> mmary. Adicional se adjunta print de pantalla de dicho resultado.



ATENTAMENTE,



Firma 1
PhD. Diego Peña Arcos
Tutor(a) metodológico



Firma 2
Mgtr. Luis Espinoza Mendoza
Tutor(a) de la ciencia

Proyecto Integrador de Ingeniería de Software Cesar Duque version final



Nombre del documento: Proyecto Integrador de Ingeniería de Software Cesar Duque version final.docx
ID del documento: d898ca6b94ed82b94998761c3f67b40735d710ab
Tamaño del documento original: 1,03 MB

Depositante: LUIS ENRIQUE ESPINOZA MENDOZA
Fecha de depósito: 6/12/2023
Tipo de carga: Interface
fecha de fin de análisis: 6/12/2023

Número de palabras: 16,288
Número de caracteres: 110,401

Ubicación de las similitudes en el documento:



Fuentes principales detectadas

Nº	Descripciones	Similitudes	Ubicaciones	Datos adicionales
1	elblogpython.com Los principios sólidos: fundamentos para un buen diseño de ... https://elblogpython.com/tecnologia/los-principios-solidos-fundamentos-para-un-buen-dise-no-de-so-... 1 fuente similar	< 1%		Palabras idénticas: < 1% (114 palabras)
2	mecambloya.com Principios SOLID En La Práctica: Código Robusto Y Mantenible https://mecambloya.com/tu-software/principios-solid-en-la-practica-codigo-robusto-y-mantenible/#...	< 1%		Palabras idénticas: < 1% (81 palabras)
3	www.ncbi.nlm.nih.gov https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10019462/pdf/psycj-14-1124095.pdf 1 fuente similar	< 1%		Palabras idénticas: < 1% (62 palabras)
4	www.doi.org https://www.doi.org/10.1016/j.CAEAI.2021.100025 12 fuentes similares	< 1%		Palabras idénticas: < 1% (37 palabras)
5	Documento de otro usuario #56a104 El documento proviene de otro grupo	< 1%		Palabras idénticas: < 1% (40 palabras)

Fuentes con similitudes fortuitas

Nº	Descripciones	Similitudes	Ubicaciones	Datos adicionales
1	integridadacademica.uc.cl https://integridadacademica.uc.cl/wp-content/uploads/2023/04/estrategias-para-fortalecer-la-integri-...	< 1%		Palabras idénticas: < 1% (38 palabras)
2	Documento de otro usuario #29679f El documento proviene de otro grupo	< 1%		Palabras idénticas: < 1% (19 palabras)
3	Documento de otro usuario #372bc6 El documento proviene de otro grupo	< 1%		Palabras idénticas: < 1% (30 palabras)
4	Documento de otro usuario #2a31b8 El documento proviene de otro grupo	< 1%		Palabras idénticas: < 1% (18 palabras)
5	Documento de otro usuario #4aaa14 El documento proviene de otro grupo	< 1%		Palabras idénticas: < 1% (29 palabras)

Fuentes mencionadas (sin similitudes detectadas)

Estas fuentes han sido citadas en el documento sin encontrar similitudes.

1	https://www.nku.edu/academic-integrity/faculty/types/index.shtml
2	https://jjay.cuny.edu/academic-dishonesty-definitions-and-examples
3	https://edubirdie.com/examples/academic-dishonesty-reasons-consequences-and-preventions/
4	https://doi.org/10.29252/johepal.1.1.25
5	https://www.scribbr.com/methodology/likert-scale/

Dedicatoria

Ada Lovelace,

A través de estas líneas, quiero expresar mi más profundo agradecimiento por tus invaluable contribuciones a la historia y la computación. Como la primera programadora en la historia, tu perspicacia y visión fueron fundamentales para sentar las bases de lo que hoy conocemos como ingeniería de software. Tu pasión por la matemática y tu creatividad para aplicarla en los algoritmos te convierten en un faro de inspiración para todos aquellos que seguimos tus pasos. Tu legado perdura y tu nombre siempre será recordado en la historia de la computación.

Alan Turing,

En este momento tan especial de mi vida, deseo rendir homenaje a tu genialidad y valentía. Tus contribuciones a la historia y la computación son inigualables. Tu trabajo en el descifrado de códigos durante la Segunda Guerra Mundial, con la invención de la máquina de Turing, marcó un hito en la computación y la inteligencia artificial. Sin duda, tu visión y tu capacidad para pensar más allá de los límites establecidos han sido y seguirán siendo una fuente de inspiración para todos los que nos dedicamos a este apasionante campo. Tu legado perdura y tu espíritu innovador siempre será un faro de luz en la comunidad científica.

Srinivasa Ramanujan,

A ti, matemático extraordinario, quiero dedicarte estas palabras llenas de admiración y respeto. A pesar de los obstáculos y las dificultades que enfrentaste en tu vida, nunca dejaste de amar las matemáticas y de perseguir ese conocimiento que tanto te apasionaba. Tus contribuciones a la teoría de los números y a la función de partición son un testimonio de tu genialidad. Tu humildad y dedicación a tu arte matemático son un ejemplo para todos nosotros, recordándonos que el amor por una disciplina y la perseverancia siempre pueden superar cualquier adversidad. Tu legado perdura y tu pasión por las matemáticas seguirá inspirando a futuras generaciones.

A estos tres grandes nombres, Ada Lovelace, Alan Turing y Ramanujan, les dedico mi tesis como un humilde tributo a su legado y como un reconocimiento a su impacto en la historia y la computación. Sin sus contribuciones, el mundo de la tecnología y la ciencia no sería el mismo.

Agradecimientos

Deseo expresar mi más profundo agradecimiento a Dios, por otorgarme el preciado don de la existencia, el amor hacia mi profesión y la habilidad que se ha convertido en mi razón de ser y vivir.

A mi madre y a mi hermana, les agradezco de todo corazón por ser testigos de mi crecimiento y pasión por la programación, desde temprana edad, cuando a los 10 años despertó en mí, el interés por las computadoras. A ti, Amanda, agradezco especialmente tu incondicional compañía, la cual me ha sostenido y motivado a no dar un paso atrás en mi camino.

Infinitamente agradezco a mi profesora del colegio, Marina Arias, quien reconoció en mí el potencial para desarrollar mis habilidades en el campo de la informática. Sus palabras de aliento y confianza fueron el impulso necesario para tomar la decisión de formarme como profesional en esta área de conocimiento y ser parte de la prestigiosa Facultad de Ingenierías.

Por último, pero no menos importante, mi profundo agradecimiento a mi docente tutor, Luis Espinoza, quien no solo me ha acompañado en el desarrollo de este trabajo de titulación de grado, sino que también ha sido un guía invaluable en mi trayectoria universitaria. Nunca olvidaré su apoyo constante y su firme convicción de que puedo superar cualquier adversidad.

A pesar de todo y quienes no creyeron en mi potencial, encontré la fuerza interior para superar mis limitaciones y demostrarles que mi éxito no depende de su aprobación, sino de mi dedicación y perseverancia.

De parvis grandis acervus erit.

Índice de Contenido

Dedicatoria.....	5
Agradecimientos	7
Índice de Contenido	8
Índice de Tablas	14
Índice de Ilustraciones	15
Resumen.....	16
Abstract.....	16
Introducción.....	17
Planteamiento del Problema	20
Objetivo General.....	21
Objetivos Específicos.....	21
Justificación	21
Capítulo 1: Marco Teórico	23
Definición de deshonestidad académica	23
Actos de deshonestidad académica en entornos virtuales de instituciones de educación superior.....	24
Prevalencia de la deshonestidad académica en el aprendizaje virtual	25
Razones del aumento de la deshonestidad académica en los entornos virtuales	26
Impacto de la deshonestidad académica	26
Esfuerzos para combatir la deshonestidad académica en entornos virtuales	26

El software asistido por inteligencia artificial para la supervisión de actividades académicas	27
Ética y responsabilidad académica	28
Importancia de la ética y la responsabilidad académica	29
Establecimiento de relaciones de confianza	29
Desarrollo personal y profesional	29
Contribución al bienestar de la comunidad.....	29
Educación a través de sistemas de información y aprendizaje en línea.....	29
Características	30
Modalidades.....	30
Herramientas.....	30
Metodologías ágiles de desarrollo de software.....	30
Feature Driven Development (FDD) o Desarrollo Basado en Características	32
Ventajas y desventajas.....	32
Importancia de la aplicación de Feature Driven Development en desarrollos de aplicaciones con inteligencia artificial.....	33
Patrones de diseño de software	33
Principios en el diseño de software.....	34
Arquitectura de software distribuido	35
Arquitectura distribuida de aplicaciones.....	37
Arquitectura REST en aplicaciones	37
Características de REST	37
Ventajas de REST	37

Desafíos de REST	38
Aprendizaje profundo	38
Redes neuronales artificiales (ANN)	39
Redes neuronales profundas	40
Aprendizaje profundo supervisado	41
Aplicaciones.....	41
Ventajas.....	41
Desventajas	41
Aprendizaje profundo en procesamiento de lenguaje natural.....	41
Modelo GPT.....	42
Aplicaciones SPA (Single Page Application).....	43
Ventajas.....	43
Desventajas	43
Tecnologías	44
Sistemas de información basados en la web apoyados en la Inteligencia Artificial.....	44
Proctoring o Supervisión en Línea de Actividades Académicas.....	45
Automatización del proctoring	45
Detección de comportamientos sospechosos.....	46
Prevención del fraude	47
Técnicas de proctoring.....	48
Capítulo 2: Marco Metodológico.....	49
Enfoque de la investigación.....	49

Definición de variables	49
Cuantitativas	49
Cualitativas	50
Alcance de la investigación	51
Correlacional.....	51
Diseño de la investigación	51
Población.....	51
Control de actividades.....	51
Técnicas de recolección de datos	52
Registros del sistema.....	52
Análisis de datos	53
Análisis de contenido.....	53
Metodología	54
Justificación	55
Implementación.....	55
Consideraciones éticas en el desarrollo y ejecución del proyecto	56
Informar consentimiento.....	56
Procedimiento	57
Capítulo 3: Análisis e Interpretación de los Resultados.....	58
Prueba de funcionalidad I	58
Prueba de funcionalidad II.....	60
Interpretación de los resultados	62

Capítulo 4: Propuesta Tecnológica	63
Descripción del software.....	63
Alcance del software.....	63
Estructura del software	63
Módulo de Cursos.....	64
Módulo de Evaluaciones.....	65
Módulo de Estudiantes.....	66
Módulo de Resolución de Evaluaciones	67
Cronograma del desarrollo del software	68
Casos de Uso.....	69
Determinación de actores.....	69
Caso 1: Ingreso a la plataforma	70
Caso 2: Supervisión de evaluaciones.....	71
Diagrama de flujo de eventos	72
Diagrama de interacción de objetos.....	72
Diagrama de arquitectura del software	77
Diseño del software.....	79
Librerías y entornos de ejecución	79
Motor de base de datos	79
Virtualización y empaquetado de la aplicación	79
Sistema operativo.....	80
IDE de desarrollo	80

API de comunicación con Inteligencia Artificial.....	80
Conclusiones	80
Recomendaciones	81
Bibliografía	82

Índice de Tablas

<i>Tabla 1: Ventajas y desventajas entre metodologías ágiles de desarrollo.....</i>	<i>32</i>
---	-----------

Índice de Ilustraciones

<i>Ilustración 1: Pruebas del sistema I - Estudiante A</i>	58
<i>Ilustración 2: Prompt para el API desde la función Typescript</i>	59
<i>Ilustración 3: Pruebas funcionales I - Estudiante B</i>	59
<i>Ilustración 4: Respuesta del API de GPT</i>	60
<i>Ilustración 5: Pruebas funcionales II - Estudiante A</i>	60
<i>Ilustración 7: Respuesta del API de GPT</i>	61
<i>Ilustración 6: Pruebas funcionales II - Estudiante B</i>	61
<i>Ilustración 8: Módulo de cursos</i>	64
<i>Ilustración 9: Módulo de evaluaciones</i>	65
<i>Ilustración 10: Módulo de estudiantes</i>	66
<i>Ilustración 11: Respuestas de estudiantes</i>	67
<i>Ilustración 12: Cronograma de actividades parte 1</i>	68
<i>Ilustración 13: Cronograma de actividades parte 2</i>	69
<i>Ilustración 14: Actores de los casos de uso</i>	70
<i>Ilustración 15: Caso de uso 1 - Ingresar al sistema</i>	70
<i>Ilustración 16: Caso de uso 2 - Evaluación</i>	72
<i>Ilustración 17: Diagrama de flujo de eventos parte 1</i>	74
<i>Ilustración 18: Diagrama de flujo de eventos parte 2</i>	75
<i>Ilustración 19: Diagrama de interacción de objetos parte 1</i>	76
<i>Ilustración 20: Diagrama de interacción de objetos parte 2</i>	77
<i>Ilustración 21: Arquitectura del software</i>	78

Resumen

El objetivo de este estudio es analizar la efectividad de este tipo de software en la supervisión de actos de deshonestidad académica en entornos virtuales de instituciones de educación superior en Ecuador. Para ello, se plantean varios objetivos específicos. En primer lugar, se busca identificar las formas prevalentes de deshonestidad académica cometidas por los estudiantes en este tipo de entornos.

Posteriormente, se propone diseñar un prototipo de software apoyado por inteligencia artificial para la generación de exámenes teóricos en un entorno de pruebas en la web. Este software permitirá la creación de evaluaciones que sean difíciles de copiar o plagiar, lo que contribuirá a prevenir actos de deshonestidad.

Una vez desarrollado el prototipo, se llevarán a cabo pruebas funcionales con estudiantes y docentes para comparar las situaciones en las que es posible el cometimiento de deshonestidad académica durante el desarrollo de estas evaluaciones. Esto permitirá evaluar la efectividad del software y detectar posibles mejoras o ajustes necesarios.

Finalmente, se evaluará la efectividad del software a través de un reporte gráfico de los resultados contrastados por el sistema frente a la revisión manual por parte de un usuario funcional. Esto permitirá determinar si el software es capaz de detectar de manera eficiente los actos de deshonestidad académica y ofrecerá información para su mejora continua.

Abstract

The objective of this study is to analyze the effectiveness of this type of software in monitoring acts of academic dishonesty in virtual environments of higher education institutions in

Ecuador. To this end, several specific objectives are proposed. First, we seek to identify the prevalent forms of academic dishonesty committed by students in this type of environment.

Subsequently, it is proposed to design a software prototype supported by artificial intelligence for the generation of theoretical exams in a web-based testing environment. This software will allow the creation of assessments that are difficult to copy or plagiarize, which will help prevent acts of dishonesty.

Once the prototype is developed, functional tests will be carried out with students and teachers to compare the situations in which academic dishonesty is possible during the development of these evaluations. This will allow us to evaluate the effectiveness of the software and detect possible improvements or necessary adjustments.

Finally, the effectiveness of the software will be evaluated through a graphical report of the results contrasted by the system versus the manual review by a functional user. This will make it possible to determine whether the software is capable of efficiently detecting acts of academic dishonesty and will provide information for its continuous improvement.

Introducción

La deshonestidad académica se refiere a diversas acciones por parte de los estudiantes que van en contra de las normas esperadas de una escuela, universidad u otra institución de aprendizaje.

Incluye el plagio, la falsificación y el engaño (Traeger & Riedel, 2009). Las causas de la deshonestidad académica son complejas y polifacéticas, e incluyen la presión por tener éxito, la falta de habilidades para gestionar el tiempo y la falta de comprensión de la integridad académica (Parnter, 2020). Las consecuencias de la deshonestidad académica pueden ser graves, desde suspender un curso hasta la expulsión de la escuela. En algunos casos, la deshonestidad académica puede incluso considerarse un delito penal (Traeger & Riedel, 2009).

Esto es un problema contemporáneo en la enseñanza superior que requiere atención urgente para ponerle freno. La cuestión de la deshonestidad académica se hizo más prominente en la segunda mitad del siglo XX, cuando las universidades adoptaron enfoques más democráticos en materia de admisión. Hoy en día, la deshonestidad académica es una característica habitual en la mayoría de los campus, si no en todos, a pesar de que se espera que la integridad académica sea un componente importante de la filosofía y los objetivos de la enseñanza superior.

Las universidades de Ecuador se enfrentan a un desafío significativo en cuanto a la deshonestidad académica, especialmente en el contexto de la enseñanza en línea. Durante la pandemia de COVID-19, la transición a la educación a distancia ha llevado a un aumento en los casos de mal comportamiento académico, como el plagio y las trampas durante los exámenes en línea. Esta situación se ha vuelto más problemática debido a la falta de interacción cara a cara en la educación en línea, lo que brinda a los estudiantes más oportunidades para participar en prácticas deshonestas con impunidad y anonimato.

El plagio es una de las prácticas deshonestas más comunes, donde los estudiantes copian y pegan información de fuentes sin atribución adecuada, con el objetivo de obtener mejores calificaciones sin esforzarse por aprender el contenido. Este comportamiento plantea una seria preocupación, ya que socava la integridad de la educación y disminuye el valor de los esfuerzos de los estudiantes honestos. Además, para los profesores resulta difícil identificar el plagio sin recurrir

a programas informáticos costosos o invertir una considerable cantidad de tiempo en la corrección de exámenes. Esto puede resultar en un trato injusto hacia los estudiantes y generar inconsistencias en las calificaciones. Asimismo, resulta complicado asegurar que los estudiantes sean evaluados en función de su comprensión real del material.

La dificultad para elaborar exámenes individuales y con un mayor grado de dificultad también es un problema relevante. Los profesores se enfrentan al desafío de crear evaluaciones que sean lo suficientemente desafiantes para evaluar el verdadero conocimiento de los estudiantes, pero al mismo tiempo, deben asegurarse de que las preguntas sean claras y comprensibles. En este sentido, el diseño de exámenes que aborden la diversidad de habilidades y conocimientos de los estudiantes puede resultar complejo y requiere un esfuerzo adicional por parte de los docentes.

A medida que las instituciones de enseñanza superior se basan cada vez más en entornos virtuales, se han desarrollado métodos más avanzados para detectar la corrupción académica.

La inteligencia artificial (IA) se ha revelado como una poderosa herramienta para detectar y abordar problemas cotidianos en las instituciones de enseñanza superior, como la deshonestidad académica. Con el auge de la educación en línea, la necesidad de medidas eficaces de supervisión y prevención se ha vuelto aún más crucial (Kumar et al., 2023). Universidades de todo el mundo han empezado a adoptar tecnología de IA para identificar trampas y plagios en exámenes y trabajos escritos. Por ejemplo, Turnitin, una popular herramienta de supervisión de plagios, ha introducido un detector de IA llamado Originality, cuyo objetivo es identificar material escrito generado por IA. El software asistido por Inteligencia artificial puede identificar y disuadir tales actos, lo que hace que este tema sea importante. La IA puede detectar problemas como el plagio, la colaboración inusual entre estudiantes y las discrepancias en los patrones de examen (Ventayen, 2023).

En respuesta a los avances tecnológicos en constante evolución, el proyecto podría contribuir a un uso más ético y responsable de esta tecnología en la enseñanza superior al analizar

cómo es su funcionamiento para supervisar con precisión la deshonestidad académica durante las evaluaciones en entornos académicos virtuales.

Planteamiento del Problema

Las universidades de Ecuador se enfrentan a un desafío significativo en cuanto a la deshonestidad académica, especialmente en el contexto de la enseñanza en línea. Durante la pandemia de COVID-19, la transición a la educación a distancia ha llevado a un aumento en los casos de mal comportamiento académico, como el plagio y las trampas durante los exámenes en línea. Esta situación se ha vuelto más problemática debido a la falta de interacción cara a cara en la educación en línea, lo que brinda a los estudiantes más oportunidades para participar en prácticas deshonestas con impunidad y anonimato.

El plagio es una de las prácticas deshonestas más comunes, donde los estudiantes copian y pegan información de fuentes sin atribución adecuada, con el objetivo de obtener mejores calificaciones sin esforzarse por aprender el contenido. Este comportamiento plantea una seria preocupación, ya que socava la integridad de la educación y disminuye el valor de los esfuerzos de los estudiantes honestos. Además, para los profesores resulta difícil identificar el plagio sin recurrir a programas informáticos costosos o invertir una considerable cantidad de tiempo en la corrección de exámenes. Esto puede resultar en un trato injusto hacia los estudiantes y generar inconsistencias en las calificaciones. Asimismo, resulta complicado asegurar que los estudiantes sean evaluados en función de su comprensión real del material.

La dificultad para elaborar exámenes individuales y con un mayor grado de dificultad también es un problema relevante. Los profesores se enfrentan al problema de crear evaluaciones que sean lo suficientemente desafiantes para evaluar el verdadero conocimiento de los estudiantes, pero al mismo tiempo, deben asegurarse de que las preguntas sean claras y comprensibles. En este sentido, el diseño de exámenes que aborden la diversidad de habilidades y conocimientos de los

estudiantes puede resultar complejo y requiere un esfuerzo adicional por parte de los docentes. Es crucial abordar este problema de manera efectiva con soluciones que promuevan la integridad académica y garanticen una evaluación justa para todos los estudiantes.

Objetivo General

- Analizar la efectividad del software apoyado por inteligencia artificial en la supervisión de actos de deshonestidad académica en entornos virtuales de instituciones de educación superior en Ecuador.

Objetivos Específicos

- Identificar las formas prevalentes de deshonestidad académica cometidas por los estudiantes en las instituciones de educación superior en entornos virtuales.
- Diseñar un prototipo de software apoyado por inteligencia artificial para la generación de exámenes teóricos en un entorno de pruebas en la web.
- Ejecutar pruebas funcionales con estudiantes y docentes para comparar las situaciones en las que es posible el cometimiento de deshonestidad académica durante el desarrollo de estas evaluaciones.
- Evaluar la efectividad del software a través de un reporte gráfico de los resultados contrastados por el sistema frente a la revisión manual por parte de un usuario funcional.

Justificación

La necesidad de abordar el problema de la deshonestidad académica en entornos virtuales en instituciones de educación superior de Ecuador. Esta problemática ha ido en aumento con el crecimiento de la educación en línea y la adopción de tecnologías digitales en el ámbito educativo. La deshonestidad académica afecta la integridad del sistema educativo y compromete la validez de las evaluaciones teóricas.

En términos de aportes teóricos, este proyecto contribuirá al conocimiento existente sobre la detección y prevención de la deshonestidad académica en entornos virtuales. A través de una

revisión exhaustiva de la literatura y la investigación empírica, se identificarán las mejores prácticas y enfoques en el uso de software asistido por IA para este propósito. Esto permitirá a los educadores y académicos tener una comprensión más sólida de las soluciones disponibles y tomar decisiones informadas para abordar este problema.

En cuanto a los aportes metodológicos, este proyecto contribuirá al bagaje técnico-instrumental y procedimental propio de la ciencia que se aborda. Se explorarán y evaluarán diferentes herramientas de software asistido por IA, analizando su efectividad y limitaciones en la detección de actos de deshonestidad académica. Además, se desarrollarán metodologías y métricas para evaluar el rendimiento de estas herramientas, lo que puede ser útil para futuras investigaciones y desarrollos en el campo.

En términos de aportes prácticos, este proyecto contribuirá a la solución de un problema práctico del contexto donde existe y se desarrolla el objeto de investigación. La implementación de software asistido por IA para la supervisión de actos de deshonestidad académica en evaluaciones teóricas en entornos virtuales puede ayudar a las instituciones de educación superior de Ecuador a garantizar la integridad de sus evaluaciones y promover un ambiente académico justo. Esto beneficiará tanto a los estudiantes, al asegurar que sus logros sean reconocidos por su propio mérito, como a los educadores, al facilitarles la identificación y abordaje de casos de deshonestidad académica.

Capítulo 1: Marco Teórico

Definición de deshonestidad académica

La deshonestidad académica, también conocida como mala conducta académica, engloba una serie de comportamientos deshonestos en el ámbito académico. Implica acciones como el engaño, el plagio, la fabricación o falsificación y el sabotaje (Academic Dishonesty Definition and Types | Academic Integrity Tutorial | Northern Illinois University, n.d.). Hacer trampas se refiere al uso de materiales o prácticas no autorizados durante los ejercicios académicos (Academic Dishonesty, Definitions and Examples, 2017). El plagio se produce cuando alguien adopta el trabajo de otra persona sin reconocerlo debidamente. La fabricación o falsificación implica la creación o alteración de información en documentos académicos. El sabotaje consiste en alterar o destruir el

trabajo de otra persona (Academic Dishonesty Definition and Types | Academic Integrity Tutorial | Northern Illinois University, n.d.).

La deshonestidad académica se considera una ofensa grave dentro de la comunidad académica, independientemente de la intención, y se aplica a todos los implicados en la enseñanza, el aprendizaje, la investigación y las actividades relacionadas (Academic Dishonesty Definition and Types | Academic Integrity Tutorial | Northern Illinois University, n.d.). Es importante que los estudiantes sean conscientes de las consecuencias de sus acciones y protejan su propio trabajo de ser utilizado de forma deshonesto. Incluso si otro estudiante utiliza su trabajo sin su conocimiento, el estudiante original puede tener alguna responsabilidad. Por lo tanto, es aconsejable que los estudiantes guarden su trabajo para sí mismos y eviten prestárselo a otros, ya que, aunque se utilice como guía, puede implicar al estudiante en deshonestidad académica si sus ideas o pasajes son utilizados por otra persona (*Quiz #2*, n.d.).

Actos de deshonestidad académica en entornos virtuales de instituciones de educación superior

La deshonestidad académica se refiere al engaño, el plagio o el comportamiento poco ético en un entorno educativo, y puede adoptar diversas formas. Los entornos de aprendizaje en línea han presentado nuevos retos y oportunidades para la deshonestidad académica. Los factores que afectan a la deshonestidad académica incluyen factores individuales, institucionales, relacionados con el medio y específicos de la evaluación (Holden et al., 2021).

Para abordar y prevenir la deshonestidad académica, las instituciones académicas cuentan con políticas y procedimientos. Estas políticas pueden incluir códigos de honor, software de detección de plagio y sistemas de supervisión de exámenes en línea (Chiang et al., 2022). Educar a los estudiantes sobre la importancia de la integridad académica y el comportamiento ético también es crucial para combatir la deshonestidad académica.

Prevalencia de la deshonestidad académica en el aprendizaje virtual

La prevalencia de la deshonestidad académica en el aprendizaje electrónico se ha convertido en una preocupación creciente en los últimos años (Malik et al., 2023). Con el rápido avance de la tecnología y la amplia disponibilidad de recursos en línea, los estudiantes tienen ahora un acceso sin precedentes a información que puede copiarse y utilizarse fácilmente sin citarla debidamente (Malik et al., 2023). Esto ha facilitado que los estudiantes cometan actos de plagio, que es una de las formas más comunes de deshonestidad académica (Malik et al., 2023).

Otra forma de deshonestidad académica que prevalece en el e-learning es hacer trampas durante los exámenes y evaluaciones online (Malik et al., 2023). La naturaleza remota del e-learning hace que sea más difícil para los instructores controlar a los estudiantes durante los exámenes, lo que lleva a un aumento de los comportamientos tramposos (Malik et al., 2023). En un estudio realizado en Pakistán, el 60% de los estudiantes admitió haber hecho trampas durante los exámenes en línea, y el 30% admitió haberlas hecho al menos una vez (Malik et al., 2023). El estudio también descubrió que hacer trampas aumentaba el rendimiento académico de los estudiantes en un 26% (Malik et al., 2023).

Además del plagio y el engaño, los entornos de aprendizaje electrónico también presentan oportunidades para otras formas de deshonestidad académica, como la fabricación o falsificación de datos (Malik et al., 2023). Las instituciones que ofrecen programas de aprendizaje electrónico deben aplicar medidas sólidas para detectar y prevenir la deshonestidad académica (Malik et al., 2023). Una revisión sistemática sobre la deshonestidad académica durante la pandemia COVID-19 identificó el apoyo del entorno y las herramientas como los factores más significativos que influyen en la deshonestidad académica en los entornos de aprendizaje en línea (Chiang et al., 2022). La revisión también proporcionó orientación a los administradores, diseñadores y desarrolladores para comprender y abordar la deshonestidad académica en los entornos de aprendizaje en línea (Chiang et al., 2022).

Razones del aumento de la deshonestidad académica en los entornos virtuales

La naturaleza remota del e-learning ha contribuido al aumento de la deshonestidad académica en los entornos virtuales. Con los cursos en línea, resulta más difícil para los profesores controlar a los estudiantes durante los exámenes y las evaluaciones, lo que conduce a una mayor prevalencia de las trampas (Holden et al., 2021). Además, la facilidad para copiar y pegar información en línea ha hecho del plagio una de las formas más comunes de deshonestidad académica en el e-learning.

Los entornos de aprendizaje en línea también ofrecen oportunidades para otras formas de deshonestidad académica, como la fabricación o falsificación de datos. La falta de interacción cara a cara y de supervisión facilita que los estudiantes manipulen datos o inventen resultados sin ser detectados (Chiang et al., 2022). Esto supone un reto importante para las instituciones que ofrecen programas de e-learning, ya que necesitan aplicar medidas para detectar y prevenir la deshonestidad académica con el fin de mantener la integridad académica.

Impacto de la deshonestidad académica

El impacto de la deshonestidad académica es de gran alcance. Socava la credibilidad de las instituciones de educación superior y devalúa los esfuerzos de los estudiantes honestos. Además, los estudiantes que incurrir en conductas deshonestas se pierden las experiencias esenciales de aprendizaje, lo que en última instancia debilita sus competencias y habilidades. Además, la deshonestidad académica erosiona la confianza dentro de la comunidad académica, lo que repercute en futuras colaboraciones y trabajos académicos.

Esfuerzos para combatir la deshonestidad académica en entornos virtuales

Muchas instituciones de enseñanza superior están tomando medidas proactivas para combatir la deshonestidad académica en los entornos virtuales. Entre ellas se incluyen el uso de software de detección de plagio, herramientas de supervisión a distancia y códigos de honor. Además, las instituciones se centran en promover la integridad académica a través de la educación,

concienciando a los estudiantes de las consecuencias de la deshonestidad académica y fomentando una cultura de honestidad y responsabilidad.

El software asistido por inteligencia artificial para la supervisión de actividades académicas

El software asistido por IA para la supervisión de exámenes en línea es una herramienta que utiliza tecnologías avanzadas de monitoreo, como el análisis de audio y video basado en inteligencia artificial, para garantizar la integridad y la seguridad durante las evaluaciones.

Algunas características y funcionalidades comunes de estos softwares de supervisión incluyen:

- **Verificación de identidad:** Estos programas utilizan métodos de verificación de identidad, como la captura de imágenes o la autenticación biométrica, para asegurarse de que el estudiante o candidato sea quien dice ser
- **Monitoreo de audio y video:** Utilizan tecnologías de inteligencia artificial para monitorear y analizar el audio y video durante el examen, detectando comportamientos sospechosos o no permitidos, como mirar hacia otro lado, hablar con alguien más o tener objetos no autorizados cerca
- **Bloqueo de aplicaciones y navegación restringida:** Estos programas pueden bloquear el acceso a otras aplicaciones o sitios web durante el examen para evitar que los estudiantes busquen respuestas o utilicen recursos no permitidos
- **Supervisión en vivo:** Algunas plataformas ofrecen la opción de tener un supervisor en vivo que vigile la sesión de examen a través de la webcam y tenga acceso a la pantalla del ordenador del estudiante para garantizar la integridad del proceso
- **Generación de informes y análisis:** Estos programas pueden generar informes detallados sobre el comportamiento y el rendimiento de los estudiantes durante el examen, proporcionando datos útiles para evaluar la integridad y la eficacia del proceso de evaluación

Algunos ejemplos de software de supervisión de exámenes en línea incluyen ExamSoft ExamMonitor, Integrity Advocate, Mettl y Proctorio. Estas herramientas ofrecen soluciones completas para la verificación de identidad, monitoreo en tiempo real, análisis de comportamiento y generación de informes.

Es importante tener en cuenta que el uso de software asistido por IA para la supervisión de exámenes en línea plantea cuestiones de privacidad y ética, por lo que es necesario implementar políticas y prácticas adecuadas para garantizar el equilibrio entre la seguridad y la protección de los derechos de los estudiantes o candidatos.

Ética y responsabilidad académica

La ética y la responsabilidad académica son cuestiones importantes en la educación, tanto para los estudiantes como para los profesores y las instituciones educativas. A continuación, se exponen algunos puntos clave sobre estas cuestiones:

- La responsabilidad académica implica que los estudiantes deben cumplir sus obligaciones académicas, como asistir a clase, entregar los trabajos a tiempo y estudiar para los exámenes. También deben demostrar una conducta ética en su comportamiento y comunicación con compañeros, profesores y asesores académicos.
- Los profesores también tienen la responsabilidad ética de ofrecer una educación de calidad y fomentar el aprendizaje de sus alumnos. Esto incluye ser justo y objetivo a la hora de evaluar a los alumnos y proporcionarles comentarios constructivos para ayudarles a mejorar.
- Las instituciones educativas también tienen la responsabilidad ética de proporcionar un entorno de aprendizaje seguro y respetuoso para todos los alumnos y profesores. También deben rendir cuentas de la calidad de la educación que imparten y garantizar que sus programas académicos sean pertinentes y estén actualizados.
- La ética y la responsabilidad académica también están relacionadas con la responsabilidad social universitaria, que implica que las instituciones educativas deben ser conscientes de su

impacto en la sociedad y trabajar para mejorarlo. Esto puede incluir la promoción de la diversidad y la inclusión, la participación en proyectos de investigación y desarrollo social y la colaboración con otras organizaciones para abordar problemas sociales y medioambientales.

Importancia de la ética y la responsabilidad académica

Establecimiento de relaciones de confianza

La ética y la responsabilidad académica son fundamentales para establecer relaciones de confianza entre los estudiantes, el profesorado y la comunidad académica en general. Actuar de forma ética y socialmente responsable crea un entorno de respeto y confianza mutuos.

Desarrollo personal y profesional

La ética y la responsabilidad académica contribuyen al desarrollo personal y profesional de los estudiantes. Actuando éticamente, los estudiantes aprenden a tomar decisiones basadas en valores y principios morales, lo que les ayuda a crecer como personas y a afrontar los retos de la vida profesional.

Contribución al bienestar de la comunidad

Actuar éticamente y con responsabilidad académica implica preocuparse por el bienestar de la comunidad y del entorno social y medioambiental. Los estudiantes y profesionales éticos contribuyen al desarrollo sostenible de la sociedad mediante prácticas sostenibles, la promoción de la igualdad y la justicia social y el fomento de la formación continua.

Educación a través de sistemas de información y aprendizaje en línea

La educación a través de sistemas de información y aprendizaje en línea es una modalidad de enseñanza que utiliza tecnologías de la información y la comunicación (TIC) para impartir conocimientos y habilidades a los estudiantes. Es un método de enseñanza educativa a través de Internet. Permite a los estudiantes obtener un título desde su casa o en cualquier lugar que se encuentren, siempre y cuando tengan acceso a una computadora con conexión a Internet.

Características

La educación en línea se caracteriza por la flexibilidad y la comodidad que ofrece a los estudiantes, ya que no es necesario estar en un aula de clases en el campus y se puede acceder fácilmente a los cursos cuando se desee. Además, los estudiantes pueden completar las actividades a su propio ritmo. Sin embargo, es necesario tener conocimientos de informática e Internet básicos, así como buenas habilidades de estudio para tener éxito en el curso de grado en línea.

Modalidades

La educación en línea puede ser ofrecida en diferentes modalidades, como la educación virtual o la educación a distancia. La educación virtual se refiere al desarrollo de programas de formación que tienen como escenario de enseñanza y aprendizaje el ciberespacio. Por otro lado, la educación a distancia se apoya en recursos digitales para desplegar los procesos de enseñanza y aprendizaje utilizando herramientas propias de las TIC.

Herramientas

Para la educación en línea se utilizan diferentes herramientas, como los sistemas de gestión de aprendizaje (LMS, por sus siglas en inglés). Los LMS son software que ofrecen programas de aprendizaje en línea en diferentes formatos, como aprendizaje formal, informal y social. Estos sistemas permiten a los instructores y alumnos completar sin problemas el proceso de aprendizaje en línea con la misma facilidad que pueden hacerlo en un aula.

En resumen, la educación a través de sistemas de información y aprendizaje en línea es una modalidad de enseñanza que utiliza tecnologías de la información y la comunicación para impartir conocimientos y habilidades a los estudiantes. Esta modalidad se caracteriza por la flexibilidad y la comodidad que ofrece a los estudiantes, y se apoya en diferentes herramientas, como los sistemas de gestión de aprendizaje.

Metodologías ágiles de desarrollo de software

Las metodologías ágiles de desarrollo de software son enfoques flexibles y colaborativos para la planificación y gestión de proyectos de software. Algunas metodologías ágiles comunes

incluyen Scrum, Kanban, Extreme Programming (XP) y Lean Software Development. A continuación, algunos puntos clave sobre las metodologías ágiles de desarrollo de software más conocidas:

- Scrum: Es una metodología ágil popular que se basa en ciclos de trabajo llamados "sprints". Durante cada sprint, el equipo se enfoca en entregar un conjunto de funcionalidades específicas. Scrum se basa en la colaboración y la comunicación constante entre los miembros del equipo y se apoya en reuniones diarias de seguimiento, revisiones de sprint y retrospectivas para mejorar continuamente el proceso.
- Kanban: Es una metodología ágil que se basa en la visualización del flujo de trabajo. Utiliza tableros Kanban para rastrear las tareas y limitar la cantidad de trabajo en progreso. Kanban se enfoca en la entrega continua y en la optimización del flujo de trabajo, permitiendo a los equipos adaptarse rápidamente a los cambios y priorizar las tareas más importantes.
- Extreme Programming (XP): Es una metodología ágil que se centra en la calidad del software y la colaboración constante entre los miembros del equipo. XP se basa en prácticas como la programación en parejas, pruebas unitarias frecuentes, integración continua y desarrollo incremental. El objetivo principal de XP es entregar software de alta calidad de manera rápida y eficiente.
- Lean Software Development: Se basa en los principios del Lean Manufacturing y se enfoca en la eliminación de desperdicios y la entrega de valor al cliente de manera eficiente. Esta metodología se centra en la optimización del flujo de trabajo, la reducción del tiempo de espera y la mejora continua. Lean Software Development promueve la colaboración, la transparencia y la toma de decisiones basada en datos.

Cada metodología tiene sus propias características y enfoques, pero comparten la idea central de adaptarse rápidamente a los cambios, fomentar la colaboración y entregar valor de manera eficiente. La elección de la metodología ágil adecuada depende de las necesidades y características del proyecto.

Feature Driven Development (FDD) o Desarrollo Basado en Características

FDD es un proceso de iteración corto y basado en modelos que se diseñó en torno a las mejores prácticas de ingeniería de software, como el modelado de objetos de dominio, el desarrollo por características, la propiedad de clases individuales y los equipos de características.

Ventajas y desventajas

Tabla 1: Ventajas y desventajas entre metodologías ágiles de desarrollo

Metodología	Ventajas	Desventajas
SCRUM	<ul style="list-style-type: none">• Fomenta la colaboración y la comunicación en equipo.• Ofrece flexibilidad para adaptarse a los cambios.• Entrega frecuente de funciones utilizables.	<ul style="list-style-type: none">• Menos predecible debido a su naturaleza flexible.• Requiere miembros experimentados del equipo para una aplicación eficaz.• No es adecuado para proyectos pequeños.
LEAN	<ul style="list-style-type: none">• Aumenta la eficiencia al eliminar los residuos.• Aumenta la transparencia de los procesos.• Fomenta la mejora continua.	<ul style="list-style-type: none">• Requiere un cambio de mentalidad y cultura.• Requiere un profundo conocimiento del flujo de valor.• Puede llevar a una optimización excesiva y a descuidar otros factores cruciales.
FDD	<ul style="list-style-type: none">• Clara visibilidad del progreso.• Eficaz para proyectos de gran envergadura.• Software de alta calidad mediante inspecciones rigurosas del código.	<ul style="list-style-type: none">• Requiere un alto nivel de documentación.• No es adecuado para equipos o proyectos pequeños.• Menos énfasis en los requisitos no funcionales.

Fuente: Elaboración propia

Importancia de la aplicación de Feature Driven Development en desarrollos de aplicaciones con inteligencia artificial

El desarrollo basado en características (FDD) como metodología de desarrollo para la creación de un sistema web apoyado en inteligencia artificial para una institución de educación superior, que debe desarrollarse en menos de 90 días, puede ofrecer varias ventajas. La principal es que se centra en ofrecer funciones tangibles y valiosas para el cliente en un plazo breve.

Dado lo limitado del plazo, la clara visibilidad de los progresos que ofrece el FDD sería muy beneficiosa para mantener a las partes interesadas al día y tranquilas. Además, el enfoque estructurado del FDD, con sus cinco actividades básicas (desarrollar un modelo general, elaborar una lista de características, planificar por características, diseñar por características y construir por características), garantiza un proceso coherente y funciones claras.

Esta estructura puede ayudar a gestionar las complejidades inherentes al desarrollo de un sistema asistido por IA. Además, la integración de las fases de diseño y construcción en el FDD favorece una transición fluida de una fase a otra, lo que aumenta la eficiencia.

Por último, la escalabilidad del FDD le permite adaptarse a cambios y modificaciones, un factor vital teniendo en cuenta la naturaleza dinámica de la IA.

Patrones de diseño de software

Los patrones de diseño de software son soluciones típicas a problemas comunes en el diseño de software. Estos patrones son modelos prefabricados que se pueden personalizar para resolver un problema de diseño recurrente en el código. Los patrones de diseño no son una pieza específica del código, sino un concepto general para resolver un problema particular. Los patrones de diseño se dividen en tres categorías: patrones de creación, patrones estructurales y patrones de comportamiento. Algunos patrones de diseño de software comunes son:

- Método fábrica: Este patrón de creación se utiliza para crear objetos sin especificar la clase exacta del objeto que se creará. En lugar de crear objetos directamente, se utiliza una fábrica para crear objetos de manera más flexible.
- Fábrica abstracta: Este patrón de creación proporciona una interfaz para crear familias de objetos relacionados o dependientes sin especificar sus clases concretas.
- Constructor: Este patrón de creación se utiliza para crear objetos complejos paso a paso. Cada paso se realiza en un método de construcción diferente.
- Prototipo: Este patrón de creación se utiliza para crear objetos a partir de un prototipo existente. El prototipo se clona y se modifica según sea necesario.
- Instancia única: Este patrón de creación se utiliza cuando una clase de un programa debe tener una única instancia para todos los clientes.
- Adaptador: Este patrón estructural se utiliza para permitir la colaboración de objetos con interfaces incompatibles.
- Puente: Este patrón estructural se utiliza para separar una abstracción de su implementación, lo que permite que ambas evolucionen de manera independiente.
- Objeto compuesto: Este patrón estructural se utiliza para tratar un grupo de objetos como si fuera un solo objeto.

Estos patrones se dividen en tres categorías: patrones de creación, patrones estructurales y patrones de comportamiento.

Principios en el diseño de software

Los principios SOLID son un conjunto de principios de diseño de software que promueven la creación de código limpio, modular y fácil de mantener. Estos principios se basan en la programación orientada a objetos y se utilizan para mejorar la calidad y la flexibilidad del software. A continuación, se presentan los cinco principios SOLID:

- Principio de Responsabilidad Única (SRP): Este principio establece que una clase debe tener una única responsabilidad y motivo para cambiar. Cada clase debe tener una única razón para existir y debe ser responsable de una sola tarea o funcionalidad.
- Principio de Abierto/Cerrado (OCP): Este principio establece que las entidades de software (clases, módulos, etc.) deben estar abiertas para su extensión pero cerradas para su modificación. Esto significa que se deben poder agregar nuevas funcionalidades sin modificar el código existente.
- Principio de Sustitución de Liskov (LSP): Este principio establece que los objetos de una clase derivada deben poder ser sustituidos por objetos de su clase base sin afectar la integridad del programa. En otras palabras, las clases derivadas deben ser completamente compatibles con las clases base.
- Principio de Segregación de la Interfaz (ISP): Este principio establece que las interfaces de una clase deben ser cohesivas y no deben tener dependencias innecesarias. Las clases no deben verse obligadas a depender de interfaces que no utilizan.
- Principio de Inversión de Dependencia (DIP): Este principio establece que los módulos de alto nivel no deben depender de módulos de bajo nivel. En cambio, ambos deben depender de abstracciones. Además, las abstracciones no deben depender de los detalles, sino que los detalles deben depender de las abstracciones.

Estos principios SOLID proporcionan pautas para escribir código limpio, modular y fácil de mantener. Al seguir estos principios, se promueve la reutilización de código, la flexibilidad y la escalabilidad del software. Los patrones de diseño también pueden ayudar a aplicar estos principios SOLID en el diseño de software.

Arquitectura de software distribuido

La arquitectura de software distribuido se refiere a la organización de los componentes de software en un sistema distribuido y cómo interactúan entre sí. Los sistemas distribuidos son sistemas de información en los cuales las funciones se reparten por áreas de trabajo diferentes que

trabajan de forma coordinada para asumir los objetivos que la organización asigna a ese sistema de información. Algunos aspectos importantes sobre la arquitectura de software distribuido son:

- Tipos de arquitecturas: Las arquitecturas de software distribuido se pueden clasificar en diferentes tipos, como on premise, híbridas o en la nube.
- Cada tipo de arquitectura tiene sus propias características y ventajas, y es importante elegir la arquitectura adecuada para el propósito del sistema.
- Componentes: Los sistemas distribuidos están compuestos por diferentes componentes, como servidores, clientes, bases de datos, aplicaciones y servicios web.

Cada componente tiene una función específica en el sistema y se comunica con otros componentes para lograr un objetivo compartido.

- Comunicación: La comunicación entre los componentes es un aspecto crítico en la arquitectura de software distribuido. Los sistemas distribuidos utilizan diferentes protocolos y tecnologías para la comunicación, como TCP/IP, HTTP, SOAP y REST.
- Es importante elegir la tecnología adecuada para la comunicación según las necesidades del sistema.
- Escalabilidad: La escalabilidad es un aspecto importante en la arquitectura de software distribuido. Los sistemas distribuidos deben ser escalables para manejar un mayor volumen de datos y usuarios.
- La arquitectura debe ser diseñada para permitir la adición de nuevos componentes y recursos según sea necesario.
- Seguridad: La seguridad es un aspecto crítico en la arquitectura de software distribuido. Los sistemas distribuidos deben ser diseñados para proteger los datos y la información del sistema de posibles amenazas y ataques. Es importante implementar medidas de seguridad adecuadas, como autenticación, autorización y cifrado de datos.

Los sistemas distribuidos utilizan diferentes tipos de arquitecturas, componentes y tecnologías de comunicación para lograr un objetivo compartido. La escalabilidad y la seguridad

son aspectos críticos en la arquitectura de software distribuido y deben ser considerados en el diseño del sistema.

La arquitectura distribuida de aplicaciones se refiere a la organización de los componentes de software en un sistema distribuido y cómo interactúan entre sí. Una técnica común utilizada en la arquitectura distribuida de aplicaciones es el uso de API REST (Representational State Transfer). A continuación, se presentan algunos aspectos importantes sobre la arquitectura distribuida de aplicaciones y API REST.

Arquitectura distribuida de aplicaciones

La arquitectura distribuida de aplicaciones se refiere a la organización de los componentes de software en un sistema distribuido y cómo interactúan entre sí. Esta arquitectura permite que los componentes se ejecuten en diferentes dispositivos y se comuniquen entre sí a través de una red.

Arquitectura REST en aplicaciones

REST es una arquitectura de software que define un conjunto de reglas para crear servicios web. Los servicios web que siguen la arquitectura REST se conocen como servicios web RESTful. Los servicios web RESTful utilizan una interfaz uniforme para acceder y manipular recursos web. La interacción en los sistemas RESTful se realiza a través del protocolo HTTP de Internet. Una API REST es una interfaz de programación de aplicaciones que utiliza la arquitectura REST para acceder y manipular recursos web.

Características de REST

Para ser una verdadera API REST, un servicio web debe cumplir con seis restricciones arquitectónicas. Estas restricciones incluyen una interfaz uniforme, un sistema cliente-servidor, sin estado, con capacidad de caché, un sistema en capas y un código bajo demanda.

Ventajas de REST

Las API REST tienen varias ventajas, como la facilidad de uso, la escalabilidad y la interoperabilidad. Son independientes del lenguaje de programación y la plataforma, lo que las hace más flexibles y adaptables.

Desafíos de REST

Los desafíos de las API REST incluyen la seguridad, la privacidad y la gestión de versiones. Es importante garantizar la seguridad y privacidad de los datos de los usuarios y abordar los posibles problemas de seguridad. También es importante gestionar las versiones de la API para garantizar la compatibilidad con las aplicaciones existentes.

En resumen, la arquitectura distribuida de aplicaciones se refiere a la organización de los componentes de software en un sistema distribuido y cómo interactúan entre sí. Las API REST son una técnica común utilizada en la arquitectura distribuida de aplicaciones y permiten el acceso y la manipulación de recursos web. Las API REST tienen ventajas como la facilidad de uso y la escalabilidad, pero también presentan desafíos como la seguridad y la gestión de versiones.

Aprendizaje profundo

El aprendizaje profundo, también conocido como deep learning, es un campo de estudio en el ámbito del aprendizaje automático que se enfoca en desarrollar algoritmos capaces de modelar abstracciones de alto nivel en conjuntos de datos. Estos algoritmos utilizan arquitecturas computacionales que permiten realizar transformaciones no lineales múltiples e iterativas en datos que se expresan en forma matricial o tensorial.

El objetivo principal del aprendizaje profundo es imitar el funcionamiento del cerebro humano al procesar información y extraer características relevantes de los datos. A diferencia de los enfoques tradicionales de aprendizaje automático, que se basan en la programación de reglas y características específicas, el aprendizaje profundo busca aprender automáticamente estas reglas y características a partir de los datos de entrada.

Una de las características clave del aprendizaje profundo es su capacidad para aprender de forma jerárquica, es decir, aprender representaciones de datos en diferentes niveles de abstracción. Esto se logra mediante el uso de redes neuronales artificiales, que consisten en capas de neuronas interconectadas. Cada capa de la red neuronal aprende características cada vez más complejas y abstractas a medida que se avanza en la jerarquía. Esta técnica se enmarca en el campo del

aprendizaje automático y emplea redes neuronales para mejorar diversas aplicaciones, como el reconocimiento de voz, la visión por ordenador y el procesamiento del lenguaje natural. Las redes neuronales tienen la capacidad de aprender a partir de datos y mejorar su rendimiento a medida que se les proporciona más información. A diferencia del aprendizaje automático tradicional, el aprendizaje profundo no requiere de un experto en el dominio para identificar la mayoría de las funciones aplicadas, ya que es capaz de comprender las características de manera incremental, eliminando así la necesidad de conocimientos especializados. Por esta razón, el aprendizaje profundo se utiliza en diversas tecnologías de inteligencia artificial, como las recomendaciones en servicios de streaming, las tecnologías de asistente de voz y la conducción autónoma.

Redes neuronales artificiales (ANN)

Las redes neuronales son un componente clave del aprendizaje profundo y se utilizan para modelar abstracciones de alto nivel en datos. Aquí hay algunos aspectos importantes sobre las redes neuronales:

- **Definición:** Las redes neuronales son algoritmos de aprendizaje automático que se basan en la estructura y funcionamiento del cerebro humano. Estas redes se componen de interconexiones de switches de red que imitan las conexiones sinápticas entre las neuronas en el sistema nervioso. El objetivo principal de las redes neuronales es aprender a reconocer patrones y realizar tareas de manera similar a como lo hace el cerebro humano.
- **Funcionamiento:** Las redes neuronales están formadas por capas de neuronas artificiales que trabajan juntas para procesar información. Cada capa procesa la información de manera diferente y se comunica con las capas adyacentes para mejorar la precisión del modelo. Las redes neuronales aprenden a partir de datos y mejoran su rendimiento a medida que se les proporciona más información.
- **Tipos:** Existen diferentes tipos de redes neuronales, como las redes neuronales convolucionales, las redes neuronales recurrentes y las redes neuronales profundas. Cada tipo de red neuronal tiene sus propias características y se utiliza para diferentes tareas,

como el reconocimiento de imágenes, el procesamiento del lenguaje natural y la predicción de series temporales.

- **Entrenamiento:** Las redes neuronales requieren un proceso de entrenamiento para aprender a partir de los datos. Durante el entrenamiento, se ajustan los pesos y los umbrales de las neuronas para mejorar la precisión del modelo. El entrenamiento puede ser supervisado, no supervisado o por refuerzo, dependiendo del tipo de tarea que se esté realizando.
- **Aplicaciones:** Las redes neuronales se utilizan en muchas aplicaciones de inteligencia artificial, como el reconocimiento de voz, la visión por ordenador, la conducción autónoma y las recomendaciones en servicios de streaming. Las redes neuronales también se utilizan en la investigación científica para modelar sistemas complejos y predecir resultados.

Redes neuronales profundas

Las redes neuronales profundas son una arquitectura tecnológica utilizada en el aprendizaje profundo o deep learning. Estas redes neuronales se basan en la estructura del cerebro humano y están formadas por capas de neuronas artificiales que trabajan juntas para procesar información y mejorar la precisión del modelo.

Las redes neuronales profundas son una de las arquitecturas tecnológicas más importantes utilizadas en el aprendizaje profundo. Estas redes neuronales se basan en la estructura del cerebro humano y están formadas por capas de neuronas artificiales que trabajan juntas para procesar información y mejorar la precisión del modelo.

Existen diferentes tipos de redes neuronales profundas, como las redes neuronales convolucionales, las redes neuronales recurrentes y las redes neuronales profundas generativas.

Las redes neuronales profundas tienen diferentes arquitecturas, como la arquitectura de red neuronal convolucional (CNN), la arquitectura de red neuronal recurrente (RNN) y la arquitectura de red neuronal generativa adversarial (GAN).

Aprendizaje profundo supervisado

El aprendizaje profundo supervisado es una técnica de aprendizaje automático que utiliza un conjunto de datos etiquetados para entrenar un modelo de red neuronal profunda.

El aprendizaje profundo supervisado es una técnica de aprendizaje automático que utiliza un conjunto de datos etiquetados para entrenar un modelo de red neuronal profunda. En el aprendizaje supervisado, el algoritmo se guía por un conocimiento previo de cuáles deben ser los valores de salida del modelo.

En el aprendizaje profundo supervisado, se utiliza un conjunto de datos etiquetados para entrenar un modelo de red neuronal profunda. Durante el entrenamiento, se ajustan los pesos y los umbrales de las neuronas para mejorar la precisión del modelo. Una vez entrenado, el modelo se puede utilizar para predecir los resultados de nuevos datos.

Aplicaciones

El aprendizaje profundo supervisado se utiliza en muchas aplicaciones de inteligencia artificial, como el reconocimiento de voz, la visión por ordenador y el procesamiento del lenguaje natural.

Ventajas

El aprendizaje profundo supervisado tiene varias ventajas, como la capacidad de manejar grandes cantidades de datos y la capacidad de mejorar la precisión del modelo a medida que se le proporciona más información.

Desventajas

El aprendizaje profundo supervisado puede requerir grandes cantidades de datos etiquetados para entrenar el modelo

Aprendizaje profundo en procesamiento de lenguaje natural

El procesamiento de lenguaje natural (NLP) es una aplicación importante del aprendizaje profundo.

El aprendizaje profundo en el procesamiento de lenguaje natural es una técnica de aprendizaje automático que utiliza redes neuronales profundas para procesar y analizar automáticamente el lenguaje humano. El objetivo es desarrollar técnicas y algoritmos para procesar y analizar automáticamente el lenguaje humano. Esta técnica se utiliza en muchas aplicaciones de inteligencia artificial y tiene ventajas como la capacidad de manejar grandes cantidades de datos y mejorar la precisión del modelo. Sin embargo, también tiene desventajas como la necesidad de grandes cantidades de datos etiquetados y la susceptibilidad al sobreajuste. También se utiliza en muchas aplicaciones de inteligencia artificial, como el reconocimiento de voz, la traducción automática, la generación de texto y la clasificación de texto.

Modelo GPT

Los modelos GPT (Generative Pre-trained Transformer) son una familia de modelos de lenguaje natural pre-entrenados desarrollados por OpenAI. Estos modelos utilizan redes neuronales profundas para procesar y analizar automáticamente el lenguaje humano. A continuación, se presentan algunos de los modelos GPT y sus características:

GPT-1: Fue el primer modelo GPT lanzado por OpenAI. Tenía 117 millones de parámetros y se entrenó en una gran cantidad de datos no etiquetados. Aunque fue un logro significativo en el procesamiento de lenguaje natural, tenía ciertas limitaciones, como la generación repetitiva de texto y la falta de cohesión en textos más largos.

GPT-2: Es un modelo GPT más grande que GPT-1, con 1.5 mil millones de parámetros. Se entrenó en una gran cantidad de datos no etiquetados y es capaz de generar textos coherentes y de alta calidad. GPT-2 también puede realizar tareas de procesamiento de lenguaje natural, como la traducción automática y la generación de texto.

GPT-3: Es el modelo GPT más grande hasta la fecha, con 175 mil millones de parámetros. Se entrenó en una gran cantidad de datos no etiquetados y es capaz de realizar tareas de procesamiento de lenguaje natural con una precisión impresionante. GPT-3 puede generar textos

coherentes y de alta calidad, y también puede realizar tareas como la traducción automática y la generación de texto.

GPT-4: GPT-4 es el último modelo de lenguaje natural pre-entrenado desarrollado por OpenAI. Este modelo utiliza redes neuronales profundas para procesar y analizar automáticamente el lenguaje humano. Aunque GPT-4 es un logro significativo en el procesamiento de lenguaje natural, todavía tiene limitaciones, como la generación de información falsa y errores de razonamiento. Se debe tener precaución al utilizar los resultados de los modelos de lenguaje natural, especialmente en contextos de alto riesgo.

Aplicaciones SPA (Single Page Application)

Las aplicaciones web de una sola página (SPA) son una modalidad de desarrollo de aplicaciones web que se caracterizan por tener todo su contenido en una sola página web, lo que permite una experiencia más fluida y rápida para el usuario.

Una SPA es una aplicación web que cabe en una sola página y que se carga dinámicamente cuando el usuario interactúa con ella. En lugar de cargar nuevas páginas enteras desde el servidor, la SPA reescribe la página en el navegador, lo que permite una experiencia más rápida y fluida.

Ventajas

Las SPA ofrecen varias ventajas, como una mayor velocidad de carga y una mejor experiencia de usuario. Al cargar solo una vez los recursos necesarios, como HTML, CSS y JavaScript, se transmiten pequeñas cantidades de datos que actualizan o reescriben la página, lo que reduce el tiempo de carga. Además, las SPA son más fáciles de desarrollar y mantener, ya que se utilizan tecnologías web como JavaScript, Angular, HTML y CSS.

Desventajas

Las SPA también presentan algunas desventajas, como la falta de compatibilidad con los navegadores antiguos y la dificultad para el posicionamiento en motores de búsqueda. Además, las SPA pueden presentar problemas de usabilidad cuando el usuario utiliza los botones de atrás y adelante del navegador.

Tecnologías

Para desarrollar una SPA se utilizan diferentes tecnologías, como JavaScript, Angular, HTML y CSS. También existen herramientas y librerías que facilitan el desarrollo de SPA, como React, Vue.js y Ember.js.

En resumen, las aplicaciones web de una sola página (SPA) son una modalidad de desarrollo de aplicaciones web que se caracterizan por tener todo su contenido en una sola página web, lo que permite una experiencia más fluida y rápida para el usuario. Las SPA ofrecen varias ventajas, como una mayor velocidad de carga y una mejor experiencia de usuario, pero también presentan algunas desventajas, como la falta de compatibilidad con los navegadores antiguos y la dificultad para el posicionamiento en motores de búsqueda.

Sistemas de información basados en la web apoyados en la Inteligencia Artificial

La Inteligencia Artificial (IA) ha revolucionado diversos ámbitos de la vida humana, y uno de los que más impacto ha tenido es el de los sistemas de información basados en la web (Chui et al., 2021). Estos sistemas, que utilizan la web como plataforma para ofrecer información y servicios, se han beneficiado enormemente de los avances en IA, mejorando la precisión, la eficiencia y la accesibilidad de la información.

La IA puede mejorar estos sistemas de varias maneras. Por ejemplo, los algoritmos de aprendizaje automático pueden analizar grandes conjuntos de datos y extraer información útil, lo que permite a los sistemas proporcionar información más precisa y relevante a los usuarios (Halevy, Norvig y Pereira, 2021). Además, los chatbots basados en IA pueden interactuar con los usuarios en lenguaje natural, proporcionando un medio más accesible y eficiente de entrega de información.

La IA también puede proporcionar una personalización avanzada en los sistemas de información basados en la web. Mediante el análisis del comportamiento del usuario, la IA puede adaptar la información y los servicios prestados a las necesidades y preferencias individuales de cada usuario (Russell & Norvig, 2021). Esta personalización puede mejorar la experiencia del usuario y aumentar la eficiencia de los sistemas.

Además, los sistemas de información basados en la web apoyados en la IA pueden beneficiarse de las capacidades de predicción de la IA. Esto puede permitir a estos sistemas anticipar las necesidades de información de los usuarios antes de que se produzcan, mejorando aún más la eficiencia y la capacidad de respuesta de los sistemas (Goodfellow, Bengio & Courville, 2021).

Proctoring o Supervisión en Línea de Actividades Académicas

Proctoring en el contexto de los exámenes en línea se refiere a la supervisión y monitoreo de los estudiantes durante la realización de los exámenes para prevenir el fraude académico. El uso de la inteligencia artificial (IA) en el proctoring ha ganado popularidad en los últimos años.

Automatización del proctoring

La educación en línea ha experimentado un crecimiento exponencial en la última década y, con ello, también ha aumentado la necesidad de garantizar la integridad académica (McCabe, Butterfield y Treviño, 2021). En respuesta a esto, se han desarrollado diversas tecnologías y estrategias, entre las que destaca la supervisión automatizada.

La supervisión automatizada se refiere al uso de tecnología inteligente para controlar y supervisar los exámenes en línea con el fin de prevenir y detectar el fraude académico (Moore, Dickson-Deane y Galyen, 2021). A diferencia de la supervisión tradicional, la supervisión automatizada no requiere la presencia física de un supervisor humano, lo que la hace práctica y eficiente para la educación en línea.

Según Karim, Kaminsky y Behrend (2021), la supervisión automatizada utiliza una combinación de tecnologías, como la inteligencia artificial (IA), el reconocimiento facial y la detección de movimiento, para garantizar la autenticidad de los estudiantes y la integridad de los exámenes. Estas tecnologías pueden detectar comportamientos sospechosos, como apartar la mirada de la pantalla durante un tiempo inusualmente largo o la presencia de una segunda persona en la sala.

A pesar de sus ventajas, la evaluación automatizada también ha sido objeto de críticas. Algunos argumentan que puede ser invasivo y potencialmente violar la privacidad de los estudiantes (Marshall, 2021). Además, puede haber problemas técnicos que afecten al rendimiento de los estudiantes, como una mala conexión a Internet o fallos en el software de supervisión (Crawford-Ferre & Wiest, 2021).

Sin embargo, la automatización de la supervisión sigue siendo una herramienta valiosa para mantener la integridad académica en la educación en línea. Como señala Watson (2021), es importante que las instituciones educativas desarrollen políticas claras y transparentes en torno al uso de estas tecnologías para garantizar que se utilicen de forma ética y justa.

Detección de comportamientos sospechosos

La educación en línea ha experimentado un crecimiento exponencial en la última década, y más aún debido a la pandemia del COVID-19. Sin embargo, este crecimiento ha ido acompañado de preocupaciones relativas a la integridad académica, especialmente en lo que se refiere a los exámenes en línea (Watson & Sottile, 2021).

Los exámenes en línea ofrecen a los estudiantes la comodidad y flexibilidad de realizarlos en cualquier momento y lugar. Sin embargo, también plantean importantes retos para los educadores, que se esfuerzan por detectar comportamientos sospechosos o deshonestos (Grijalva, Kerkvliet & Nowell, 2021).

Una práctica común en los exámenes en línea es el uso de tecnología de supervisión o verificación a distancia. A pesar de su eficacia en la disuasión del plagio y otras formas de engaño, ha sido objeto de críticas debido a la preocupación por la privacidad de los estudiantes y la falta de coherencia en su aplicación (Lang, 2021).

La inteligencia artificial (IA) y el aprendizaje automático también se están utilizando en la detección de comportamientos sospechosos. Algunas plataformas están diseñadas para analizar patrones de comportamiento, como la velocidad de respuesta y la frecuencia de visitas a páginas web específicas, con el fin de identificar posibles irregularidades (Robles, 2021).

Además, algunas investigaciones sugieren que los exámenes en línea podrían diseñarse de forma que disuadan de comportamientos sospechosos. Los exámenes de libro abierto, por ejemplo, pueden reducir la tentación de los estudiantes de buscar respuestas en línea porque se les permite hacerlo abiertamente (McCabe, Treviño y Butterfield, 2021).

A pesar de estos avances, detectar comportamientos sospechosos en los exámenes en línea sigue siendo un reto. La educación en línea es un campo en rápida evolución, y es probable que la necesidad de garantizar la integridad académica siga impulsando la innovación en este espacio (Watson & Sottile, 2021).

Prevención del fraude

La prevención del fraude académico en los exámenes en línea es una cuestión que requiere un enfoque global, ya que afecta no sólo a la calificación de los estudiantes, sino también a la integridad y la seguridad de los profesionales (Bunn, Caudill y Gropper, 1992). La incidencia de este tipo de fraude se ha hecho más prominente con el auge de la educación en línea, por lo que resulta crucial abordar este problema con eficacia.

Un estudio reciente demostró que los estudiantes consideran el fraude académico como una práctica válida y aceptada en el proceso académico (Bunn, Caudill, & Gropper, 1992). Esto subraya la necesidad de educar a los estudiantes sobre la importancia de la integridad académica y las consecuencias de no respetarla.

El uso de las normas APA para escribir y hacer citas y referencias es una herramienta vital para garantizar la autenticidad y la integridad en la comunicación académica (Explicar la importancia de utilizar las normas APA para escribir y hacer citas y referencias, 15%). Estas normas proporcionan un marco para estructurar nuestros trabajos de investigación y atribuir correctamente las fuentes de información.

Existen diferentes medidas para hacer frente a posibles incidencias en las pruebas de evaluación o elaboración (aunque el citado reglamento recoge en su articulado diferentes medidas para hacer frente a posibles incidencias en las pruebas de evaluación o elaboración...). Estas

medidas pueden incluir la monitorización online, el uso de software de detección de plagio y la aplicación de estrictas políticas antifraude.

Un estudio realizado en cuatro prestigiosas universidades colombianas reveló que más del 90% de los estudiantes admitían haber cometido fraude académico (Bunn, Caudill y Gropper, 1992). Esta alarmante estadística pone de relieve la importancia de tomar medidas preventivas y punitivas contra el fraude académico.

El texto también menciona la correlación entre altos niveles de corrupción y fraude académico (Bunn, Caudill y Gropper, 1992). Esto sugiere que es necesario un cambio cultural y ético, así como la aplicación de medidas preventivas, para combatir el fraude académico.

Técnicas de proctoring

El proctoring es una técnica utilizada para supervisar y controlar los exámenes en línea, asegurando la integridad y evitando el fraude académico. Aquí se presentan algunas técnicas comunes de proctoring:

- **Proctoring visual:** Esta técnica utiliza cámaras de video para monitorear a los estudiantes durante el examen. Las cámaras pueden estar integradas en el dispositivo del estudiante o pueden ser cámaras externas. El proctoring visual permite a los supervisores observar las acciones y el comportamiento de los estudiantes durante el examen.
- **Proctoring de audio:** Esta técnica implica el uso de micrófonos para capturar el audio durante el examen. Permite a los supervisores escuchar cualquier conversación o sonido sospechoso que pueda indicar un comportamiento fraudulento.
- **Proctoring de pantalla:** Esta técnica permite a los supervisores ver la pantalla del estudiante durante el examen. Puede ayudar a detectar el uso de dispositivos no permitidos, como navegadores web o aplicaciones externas.
- **Proctoring de reconocimiento facial:** Esta técnica utiliza algoritmos de reconocimiento facial para verificar la identidad de los estudiantes durante el examen. Puede comparar la

imagen en tiempo real del estudiante con una imagen de referencia para garantizar que no haya suplantación de identidad.

- Proctoring de detección de trampas: Esta técnica utiliza algoritmos de inteligencia artificial para detectar comportamientos sospechosos durante el examen, como mirar hacia otro lado, movimientos inusuales o cambios en el patrón de escritura. Puede ayudar a identificar posibles intentos de fraude.

Capítulo 2: Marco Metodológico

Enfoque de la investigación

Se utilizó un enfoque metodológico cualitativo y cuantitativo en conjunto, lo que permitió una comprensión y conocimiento más profundo de la realidad que se está investigando.

Definición de variables

Cuantitativas

- Número de Evaluaciones Virtuales: Esta variable cuantifica el número total de evaluaciones teóricas realizadas utilizando entornos virtuales en la Universidad ECOTEC.
- Número de Incidentes de Deshonestidad Académica: Recuento de casos en los que la deshonestidad académica es detectada por el software asistido por IA durante las evaluaciones virtuales.
- Precisión de la detección de IA: Medida cuantitativa de la capacidad del software para identificar correctamente incidentes de deshonestidad académica, normalmente expresada en porcentaje.

- Ahorro de tiempo: La cantidad de tiempo que se ahorra utilizando el software de IA en comparación con la supervisión manual en la detección de la deshonestidad académica.
- Puntuaciones de los estudiantes: Calificaciones o puntuaciones cuantitativas de los estudiantes con respecto a su satisfacción y percepción de imparcialidad del uso del software de IA para las evaluaciones, estas valoraciones serán tomadas al finalizar la evaluación.

Cualitativas

- Percepción de los estudiantes: Recopilar datos cualitativos mediante encuestas o entrevistas para comprender cómo perciben los estudiantes el uso del software de IA para la evaluación y si lo consideran justo.
- Comentarios de los supervisores: Obtener comentarios de los supervisores sobre sus experiencias y observaciones al utilizar software de IA para la supervisión.
- Experiencia del usuario: Explorar los aspectos cualitativos de la experiencia del usuario, incluida la facilidad de uso, la facilidad de uso y cualquier desafío al que se enfrenten tanto los estudiantes como los supervisores.
- Consideraciones éticas: Investigar las consideraciones éticas y los posibles problemas relacionados con la privacidad, la parcialidad y la imparcialidad en la evaluación asistida por IA.
- Análisis coste-beneficio: Evaluar cualitativamente si los beneficios del uso de software de IA, como la mejora de la eficiencia o la reducción de incidentes de deshonestidad académica, compensan los costes y los posibles inconvenientes.
- Comparación con la supervisión manual: Comparar cualitativamente las experiencias y los resultados del uso de software de IA con los de la supervisión manual tradicional.
- Recomendaciones de mejora: Recopilar información cualitativa sobre las áreas en las que el software asistido por IA podría mejorarse u optimizarse.

Alcance de la investigación

Correlacional

Este estudio adopta un tipo de investigación correlacional con el objetivo principal de dilucidar los atributos asociados con un fenómeno particular o una cohorte específica de individuos. En el contexto de este esfuerzo de investigación, emplearemos una metodología de encuesta y entrevista para evaluar la correlación entre la utilización de software asistido por IA para monitorear los incidentes de deshonestidad académica dentro de entornos virtuales de instituciones de educación superior en Ecuador y el conocimiento que existe por parte de los agentes que intervienen en esta problemática.

El instrumento de la encuesta incluye preguntas relativas a la frecuencia de los casos de deshonestidad académica, la eficacia del software de control, así como las perspectivas de los estudiantes y los profesores en relación con su aplicación.

Diseño de la investigación

En esta sección se abarca el cómo se obtiene información sobre la experiencia de los usuarios de la usabilidad de una aplicación web en una Institución de Educación Superior. Para lograr esto, se utilizarán técnicas de recolección de datos, como entrevistas con los administradores del sistema y el registro y seguimiento de actividades administrativas en el sistema.

Población

Institución de Educación Superior de la Zona 8 de la provincia del Guayas, Ecuador

Control de actividades

Kanban es una herramienta popular utilizada en el desarrollo de proyectos ágiles. A continuación, se presentan algunos aspectos importantes sobre Kanban como una herramienta para el control de las actividades en el desarrollo de proyectos ágiles:

- Se enfoca en la comunicación en tiempo real de la capacidad y la transparencia total del trabajo. Los elementos de trabajo se representan visualmente en un tablero Kanban, lo que

permite a los miembros del equipo ver el estado de cada pieza de trabajo en cualquier momento.

- Se utiliza para visualizar el flujo de trabajo y limitar el trabajo en progreso. Esto ayuda a los equipos a identificar cuellos de botella y a mejorar la eficiencia del proceso de desarrollo.
- Se adapta bien a proyectos de cualquier tamaño y complejidad. Es especialmente útil para proyectos que tienen requisitos cambiantes o que requieren una entrega rápida.
- Se enfoca en la entrega de funcionalidades pequeñas e incrementales a lo largo del ciclo de desarrollo. Esto permite a los equipos de desarrollo mantener la consistencia y el control sobre el producto final.
- Puede combinar con otras metodologías ágiles, como Scrum, para crear una metodología híbrida llamada "Scrumban". Esto permite a los equipos aprovechar las fortalezas de ambas metodologías.

Técnicas de recolección de datos

Registros del sistema

Se espera obtener información tanto de los administradores del sistema como de los usuarios finales (instructores y estudiantes), en los respectivos registros una vez terminadas las pruebas funcionales con cada rol de usuario respectivamente.

Método

- Entrevistas con los administradores del sistema.
- Registro y seguimiento de actividades administrativas en el sistema.

Instrumentos

- Guía de entrevistas para obtener información cualitativa sobre la experiencia del usuario, desafíos percibidos y sugerencias de mejora.
- Registro automático de actividades administrativas realizado por el sistema (logs).
- Recopilación de comentarios y sugerencias directamente desde la interfaz de la aplicación.

Procedimiento

Analizar los registros automáticos del sistema para obtener datos cuantitativos sobre el uso y las interacciones de los administradores.

Aspectos Éticos

Garantizar la privacidad y confidencialidad de la información recopilada.

Análisis de datos

Objetivo

Resumir y describir las principales características de un conjunto de datos.

Técnicas

Calcular la media, la mediana y la moda de las valoraciones de los usuarios respecto a la usabilidad de la aplicación web.

Análisis de contenido

Objetivo

Analizar sistemáticamente el contenido de los datos textuales.

Técnicas

Categorizar y analizar datos cualitativos procedentes de comentarios de usuarios dentro de la interfaz de la aplicación para identificar problemas comunes o aspectos positivos comunicados por los usuarios.

Preparación de los datos

- Limpiar y organizar los conjuntos de datos cuantitativos y cualitativos.
- Convertir los datos cualitativos en un formato adecuado para el análisis (por ejemplo, codificar las respuestas abiertas).

Análisis cuantitativo

- Calcular las estadísticas descriptivas de las preguntas de la encuesta cuantitativa.

- Realizar estadísticas inferenciales para probar hipótesis o identificar patrones.

Análisis cualitativo

- Realizar análisis temáticos para identificar temas clave en los datos cualitativos.
- Utilizar el análisis de contenido para categorizar y analizar la información textual.

Integración: Correlacionar los resultados de los análisis cuantitativos y cualitativos para proporcionar una comprensión global de las experiencias de los usuarios, durante del sistema.

Metodología

Para el proyecto se utilizará una metodología de desarrollo de software ágil, la metodología Feature Driven Development y el uso de la herramienta Kanban para la administración de actividades.

El desarrollo basado en características (FDD) como metodología de desarrollo para la creación de un sistema web apoyado en inteligencia artificial para una institución de educación superior, que debe desarrollarse en menos de 90 días, puede ofrecer varias ventajas. La principal es que se centra en ofrecer funciones tangibles y valiosas para el cliente en un plazo breve.

Dado lo limitado del plazo, la clara visibilidad de los progresos que ofrece el FDD sería muy beneficiosa para mantener a las partes interesadas al día y tranquilas. Además, el enfoque estructurado del FDD, con sus cinco actividades básicas (desarrollar un modelo general, elaborar una lista de características, planificar por características, diseñar por características y construir por características), garantiza un proceso coherente y funciones claras.

Esta estructura puede ayudar a gestionar las complejidades inherentes al desarrollo de un sistema asistido por IA. Además, la integración de las fases de diseño y construcción en el FDD favorece una transición fluida de una fase a otra, lo que aumenta la eficiencia.

Por último, la escalabilidad del FDD le permite adaptarse a cambios y modificaciones, un factor vital teniendo en cuenta la naturaleza dinámica de la IA.

Justificación

- **Flexibilidad y adaptabilidad:** Las metodologías ágiles son conocidas por su flexibilidad y adaptabilidad. En el contexto del desarrollo de un sistema complejo como un sistema de control de exámenes, los requisitos pueden evolucionar a medida que las partes interesadas comprenden mejor las necesidades del sistema. La metodología ágil permite el desarrollo iterativo y la reevaluación frecuente de las prioridades.
- **Participación y comentarios de los usuarios:** Las metodologías ágiles priorizan la colaboración constante con las partes interesadas, incluidos los usuarios finales. Dado que el sistema de supervisión implica a administradores, instructores y estudiantes, es crucial contar con el feedback continuo de estos diversos grupos de usuarios. Las ceremonias ágiles como la revisión de sprints y las reuniones diarias facilitan la retroalimentación regular, asegurando que el sistema se alinea con las expectativas de los usuarios.
- **Mitigación de riesgos e incertidumbres:** El desarrollo de un sistema de supervisión de exámenes basado en la web implica varios retos técnicos y logísticos. El enfoque incremental e iterativo de Agile permite identificar y mitigar los riesgos en una fase temprana del proceso de desarrollo. Las retrospectivas periódicas ofrecen oportunidades para abordar los problemas y mejorar continuamente el proceso de desarrollo.
- **Arquitectura y diseño evolutivos:** Agile promueve la planificación adaptativa y el desarrollo evolutivo. Esto es particularmente beneficioso para un sistema que puede requerir ajustes continuos basados en prácticas educativas cambiantes, consideraciones de seguridad o avances tecnológicos. Un enfoque ágil se adapta a la naturaleza evolutiva del panorama educativo.

Implementación

Para aplicar la metodología Feature-Driven Development (FDD) al desarrollo de la aplicación web, se realizarán los siguientes pasos:

1. Desarrollar el modelo general: Redactar un esquema para definir el modelo de entidades que intervienen en el proceso.
2. Elaborar una lista de características: Identificar y enumerar las características que deben desarrollarse para la aplicación web.
3. Planificación por características: Planificar el proceso de desarrollo dividiendo las funciones en tareas más pequeñas y asignándole a los equipos de desarrollo.
4. Diseño por características: Diseñar las funciones en función de los requisitos y las necesidades de los usuarios.
5. Construcción por características: Desarrollar las funciones en iteraciones periódicas y predecibles, centrándose en las necesidades del proyecto, las cuales se resumen en los siguientes módulos:
 - Evaluaciones
 - Cursos
 - Banco de preguntas para exámenes
 - Estudiantes
 - Usuarios
 - Desarrollo de evaluaciones

Consideraciones éticas en el desarrollo y ejecución del proyecto

Informar consentimiento

Garantizar que todos los participantes en el estudio son plenamente conscientes de la naturaleza, la finalidad y los posibles riesgos de su participación y dan su consentimiento voluntario e informado.

Procedimiento

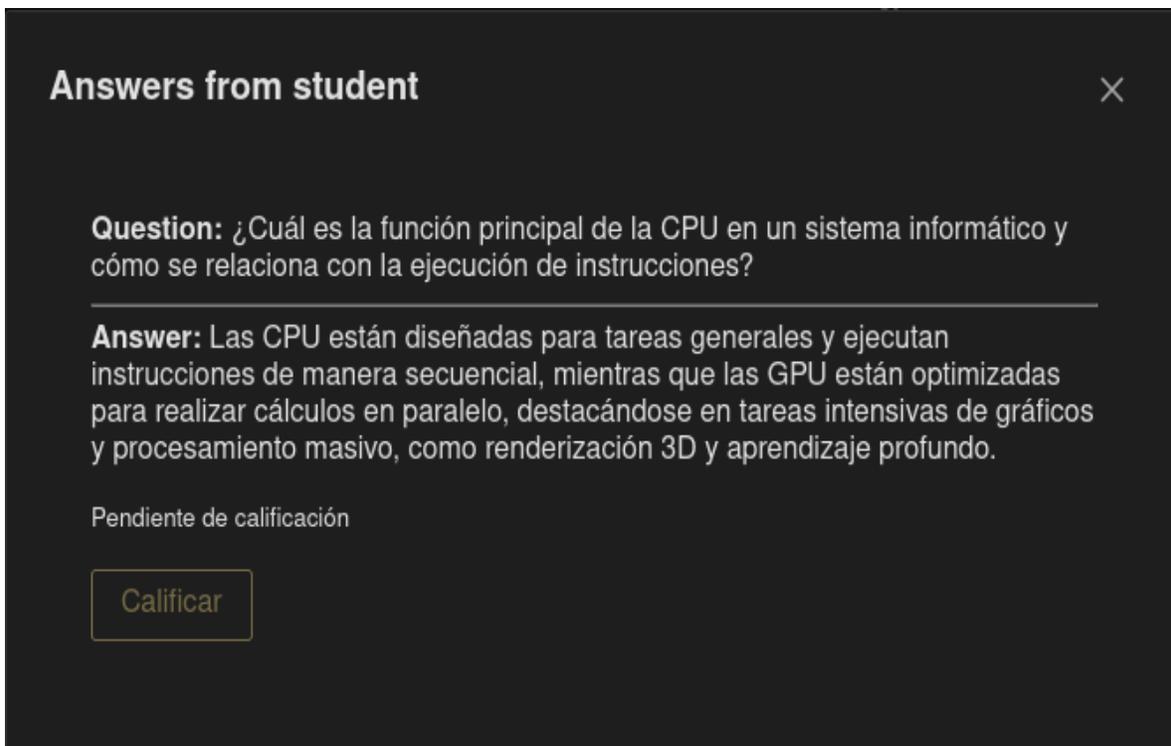
- Hoja informativa: Antes de la participación, proporcionar una hoja informativa detallando el propósito del estudio, los procedimientos implicados, los riesgos potenciales y los beneficios.
- Explicación verbal: Realice una explicación verbal del estudio, permitiendo a los participantes hacer preguntas y solicitar aclaraciones.
- Formulario de consentimiento: Distribuya un formulario de consentimiento en el que se describen los detalles del estudio y se indique explícitamente que la participación es voluntaria.
- Confirmación: Obtenga la firma de los participantes en el formulario de consentimiento para confirmar su participación informada y voluntaria.

Capítulo 3: Análisis e Interpretación de los Resultados

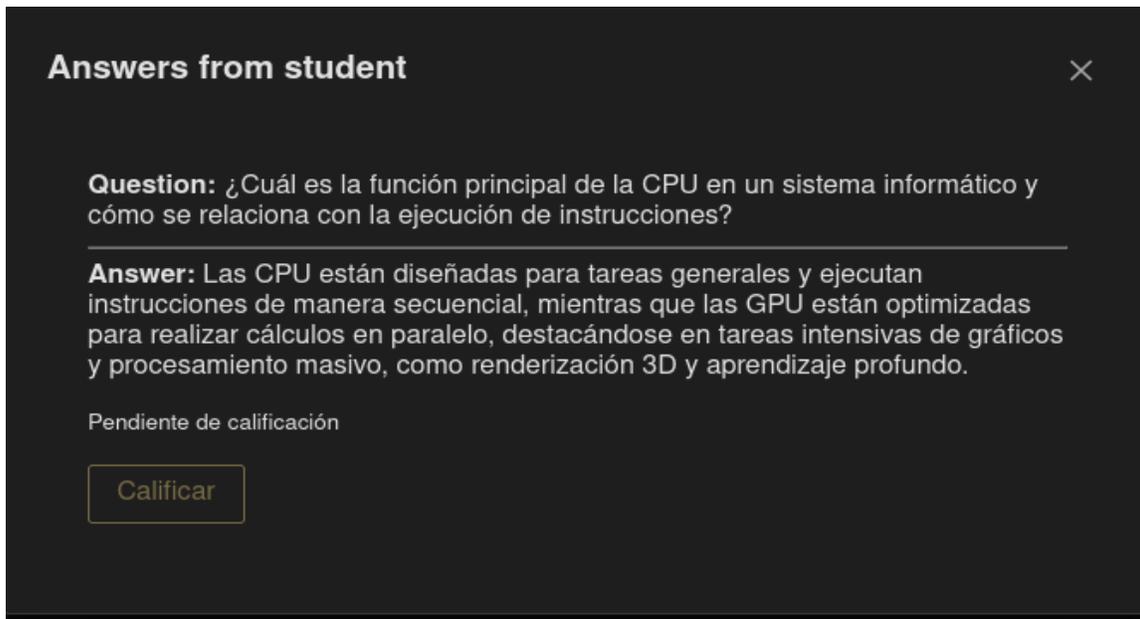
En este capítulo se exponen los resultados derivados de la ejecución del proceso de análisis y comparación de las respuestas de los estudiantes, obtenidas mediante la aplicación de funciones de generación de datos aleatorios, en el contexto de un mismo examen. Este procedimiento se llevó a cabo a través de la utilización del API de GPT como una herramienta de comunicación integrada en el sistema desarrollado, con el objetivo de brindar soporte a las decisiones que el supervisor (docente) tome en relación con las calificaciones. De esta forma, se busca validar la pertinencia y efectividad de este proyecto.

Prueba de funcionalidad I

Ilustración 1: Pruebas del sistema I - Estudiante A



Fuente: Captura del sistema



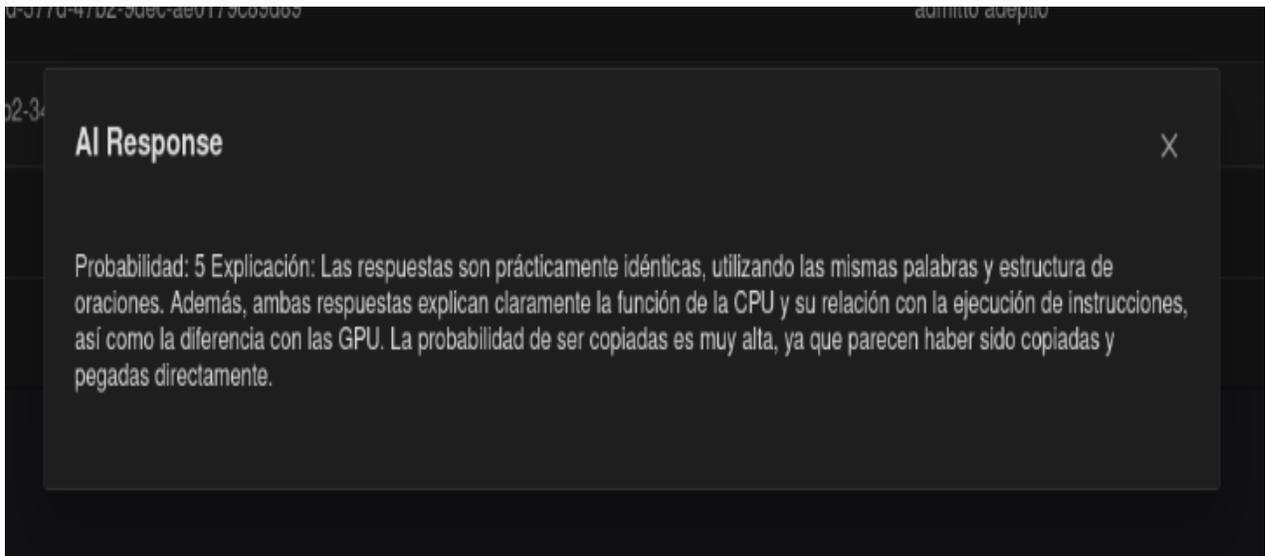
Fuente: Captura del sistema

Ilustración 2: Prompt para el API desde la función Typescript

```
showResponseAI = false
messageAI: any = ''
showResponseAILoading = false
async compareAnswers(students: any){
  this.showResponseAILoading = true
  const answersArray = this.getAnswersFromArr(students)
  const question = this.getQuestion(students)
  await this.apiService.connectToChatGPT(`Para esta pregunta: ${question}, tengo estas respuestas a continuación,
Indicame en una escala del 1 al 5 la probabilidad de que se hayan copiado estas respuestas a continuación
y explicame brevemente el porqué: ${answersArray}`).then((res)=>{
  this.messageAI = res[0]
}).finally(()=>{
  this.showResponseAI = true
  this.showResponseAILoading = false
})
}
```

Fuente: Captura del sistema

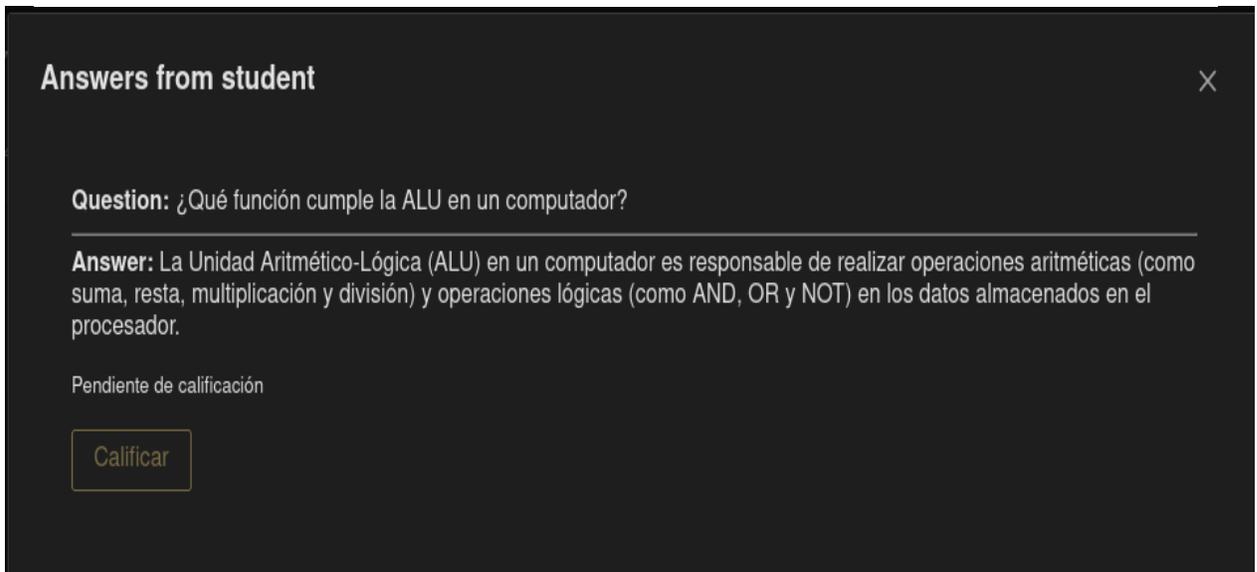
Ilustración 4: Respuesta del API de GPT



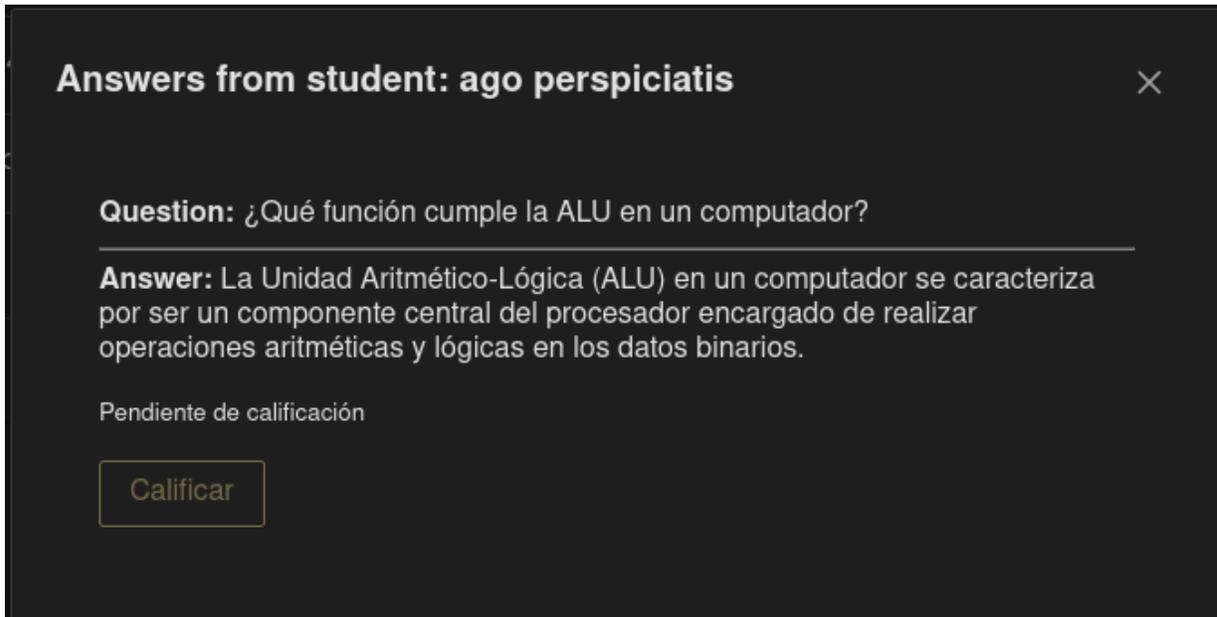
Fuente: Elaboración propia

Prueba de funcionalidad II

Ilustración 5: Pruebas funcionales II - Estudiante A



Fuente: Captura del sistema



Answers from student: ago perspiciatis ✕

Question: ¿Qué función cumple la ALU en un computador?

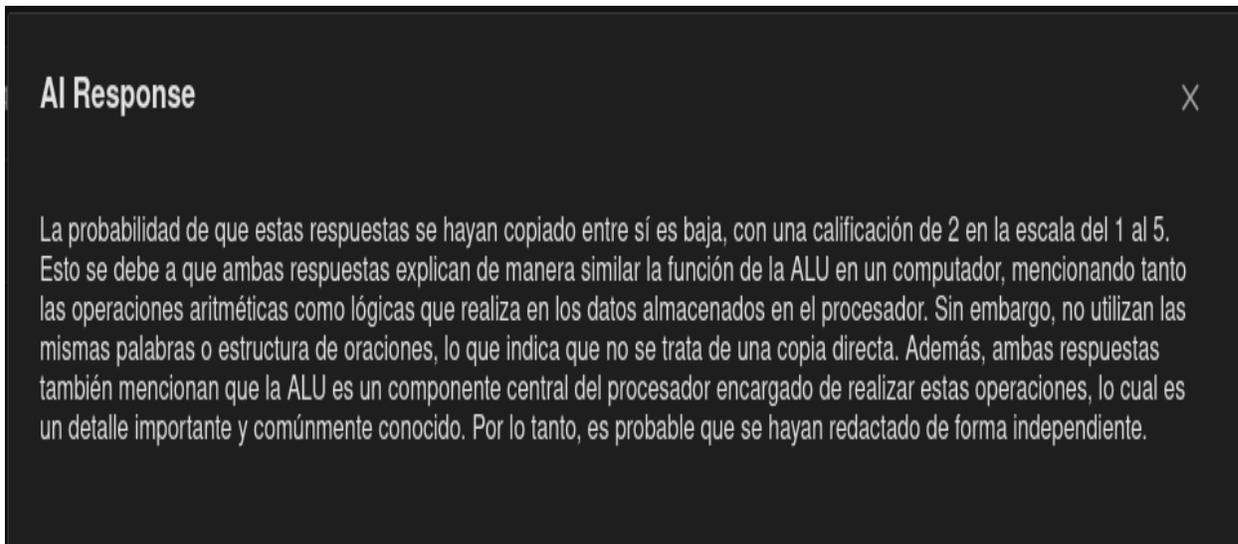
Answer: La Unidad Aritmético-Lógica (ALU) en un computador se caracteriza por ser un componente central del procesador encargado de realizar operaciones aritméticas y lógicas en los datos binarios.

Pendiente de calificación

[Calificar](#)

Fuente: Captura del sistema

Ilustración 6: Respuesta del API de GPT



AI Response ✕

La probabilidad de que estas respuestas se hayan copiado entre sí es baja, con una calificación de 2 en la escala del 1 al 5. Esto se debe a que ambas respuestas explican de manera similar la función de la ALU en un computador, mencionando tanto las operaciones aritméticas como lógicas que realiza en los datos almacenados en el procesador. Sin embargo, no utilizan las mismas palabras o estructura de oraciones, lo que indica que no se trata de una copia directa. Además, ambas respuestas también mencionan que la ALU es un componente central del procesador encargado de realizar estas operaciones, lo cual es un detalle importante y comúnmente conocido. Por lo tanto, es probable que se hayan redactado de forma independiente.

Fuente: Captura del sistema

Interpretación de los resultados

La escala de Likert no suele utilizarse para evaluar o comparar directamente similitudes entre dos o más textos. Se utiliza principalmente para evaluar cuantitativamente opiniones, actitudes o comportamientos (Bhandari, 2023). Sin embargo, la escala de Likert puede adaptarse para recoger datos sobre percepciones o preferencias relacionadas con los textos.

En las pruebas de funcionalidad A y B, realizadas conforme a las ilustraciones de la Prueba de Funcionalidad I, se llevó a cabo una comparación entre dos respuestas correspondientes a una misma pregunta. Se observó que las respuestas A y B mostraron similitudes significativas, ya que la respuesta B fue un parafraseo del contenido de la respuesta A. Para evaluar esta similitud, se utilizó el análisis con GPT, el cual determinó, en una escala de Likert del 1 al 5 (siendo 1 la calificación más baja y 5 la más alta), que la similitud entre ambas respuestas fue calificada con un 5.

Por otro lado, en la Prueba de Funcionalidad II se realizó una comparación entre dos respuestas a una misma pregunta. Se encontró que las respuestas A y B presentaron diferencias significativas, ya que la respuesta B describió características adicionales al tema propuesto en la pregunta, a diferencia de la respuesta A. Para evaluar esta diferencia, se utilizó el análisis con GPT y se determinó que, en la escala de Likert del 1 al 5, la similitud entre ambas respuestas fue calificada con un 2.

Estas pruebas proporcionan al docente una herramienta más precisa y eficiente para inferir calificaciones más precisas a cada estudiante, al facilitar la evaluación de los resultados de manera más efectiva en términos de tiempo y precisión.

Capítulo 4: Propuesta Tecnológica

Descripción del software

El software propuesto se define como un sistema web que permite a los supervisores (docentes) acceder a través de un cliente gráfico en el navegador a un curso específico y crear evaluaciones teóricas de opción múltiple. Además de establecer detalles como preguntas, puntaje total, puntaje por pregunta, respuestas válidas y opciones de respuesta. El software también integra una conexión con el API de GPT-4 para realizar un análisis inmediato de las respuestas proporcionadas por los estudiantes en comparación con las respuestas de otros estudiantes.

Alcance del software

- Creación de evaluaciones teóricas: El sistema permitirá a los supervisores (docentes) crear evaluaciones teóricas de opción múltiple, donde podrán especificar preguntas, puntajes, respuestas válidas y opciones de respuesta.
- Interfaz de administración: Se desarrollará una interfaz de administración basada en Strapi, que facilitará la gestión de usuarios, evaluaciones y registros de deshonestidad académica.
- Análisis de respuestas: El software utilizará el API de GPT-4 para comparar las respuestas proporcionadas por los estudiantes con las respuestas de otros estudiantes, identificando posibles indicios de deshonestidad académica.
- Generación de informes: El sistema generará informes detallados que resalten las respuestas sospechosas o que presenten similitudes significativas con respuestas de otros estudiantes, para facilitar la revisión y toma de decisiones por parte de los supervisores.

El alcance del proyecto se centrará en estas funcionalidades específicas, dejando espacio para ampliar y mejorar el sistema en futuras etapas de desarrollo.

Estructura del software

El software propuesto se organiza en módulos que permiten una gestión eficiente de los cursos, evaluaciones y estudiantes involucrados en el sistema de supervisión de deshonestidad

académica en entornos virtuales. A continuación, se detallan los módulos y submódulos que conforman la estructura del software.

Módulo de Cursos

Este módulo presenta una interfaz en forma de tabla que muestra los registros de los cursos asignados a un supervisor. El supervisor tiene la capacidad de modificar y visualizar en detalle las evaluaciones y estudiantes asociados a cada curso. Dentro de cada curso, se pueden diseñar evaluaciones, calificar las respuestas y visualizar los resultados de cada una de ellas.

Ilustración 8: Módulo de cursos

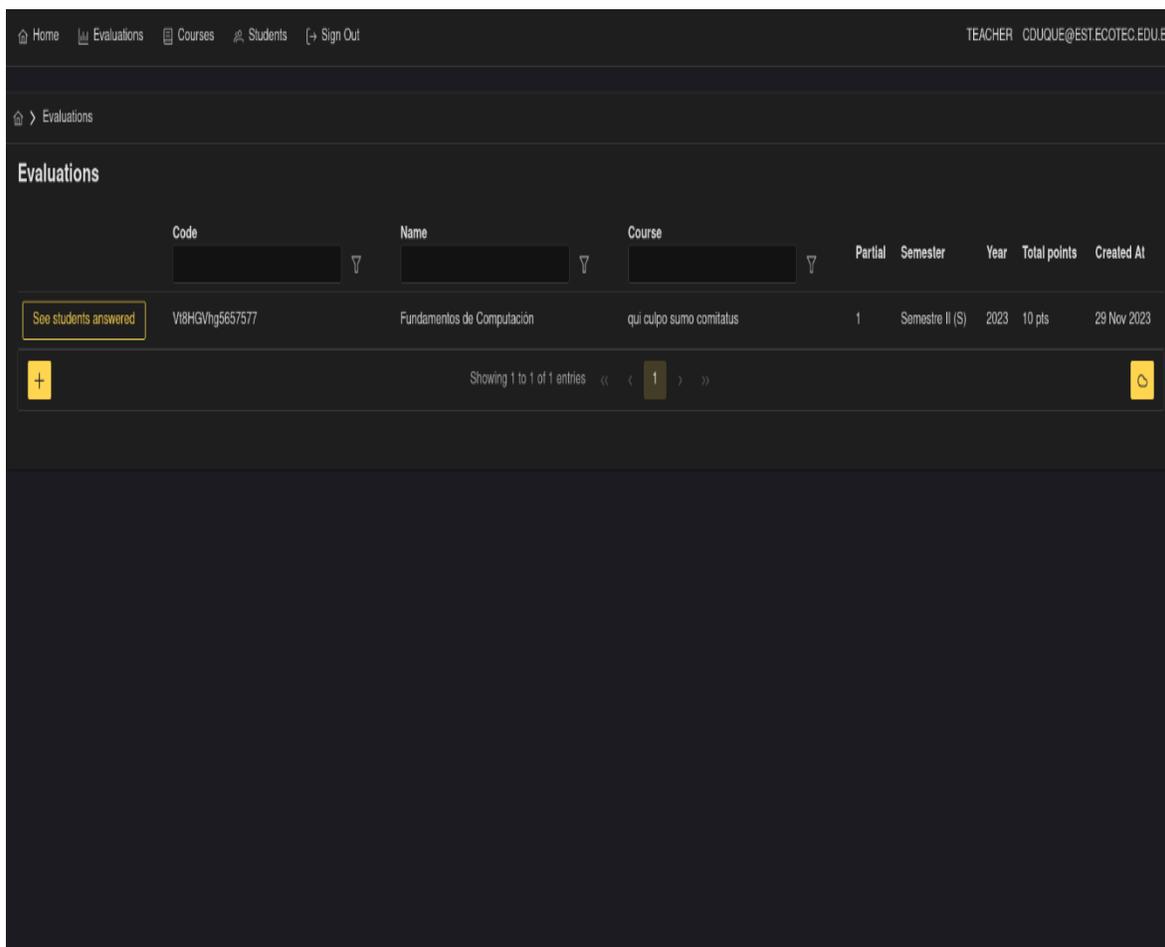
Code	Name	Description	Created At
Admoveo	qui culpa sumo comitatus	Reiciendis	20 Nov 2023

Fuente: Elaboración propia

Módulo de Evaluaciones

En este módulo se presentan todas las evaluaciones correspondientes al curso seleccionado. El supervisor tiene la posibilidad de actualizar los detalles de cada evaluación, iniciar las evaluaciones, calificarlas y visualizar las respuestas proporcionadas por los estudiantes. Este submódulo permite una gestión integral de las evaluaciones, asegurando un seguimiento completo y detallado de cada una de ellas.

Ilustración 9: Módulo de evaluaciones



The screenshot shows a web interface for the 'Evaluations' module. At the top, there is a navigation bar with links for Home, Evaluations, Courses, Students, and Sign Out. The user is identified as 'TEACHER CDUQUE@EST.ECOTEC.EDU.EC'. Below the navigation bar, the page title is 'Evaluations'. The main content area features a table with the following columns: Code, Name, Course, Partial, Semester, Year, Total points, and Created At. A single entry is displayed in the table, with a 'See students answered' button next to the Code field. Below the table, there is a pagination control showing 'Showing 1 to 1 of 1 entries' and a page number '1'.

Code	Name	Course	Partial	Semester	Year	Total points	Created At
V18HG3Vhg5657577	Fundamentos de Computación	qui culpa sumo comitatus	1	Semestre II (S)	2023	10 pts	28 Nov 2023

Fuente: Elaboración propia

Módulo de Estudiantes

Este módulo muestra en forma de tabla los registros de todos los estudiantes inscritos en el curso. Se ofrece la capacidad de filtrar los registros, actualizar la información de los estudiantes y visualizar en detalle la información de un estudiante seleccionado. Esta funcionalidad facilita la gestión de los estudiantes y proporciona una visión completa de los participantes en el curso.

Ilustración 10: Módulo de estudiantes

Code	Names	Second Names	Faculty	Career
a95a5b2f-2968-40db-b93e-114417024c62	undique quod	laborum compello	Attonbitus	Confero
fabf2298-d497-4918-b632-27c4f8ddb278	vigilo creta	solus eaque	Fugit	Votum
0eb2901b-4a6e-4968-8e98-6a1309f589a6	tergum umquam	curiositas caute	Fugit	Spectaculum
181e17d5-8cdc-41fc-8c52-7da60651f195	taedium denuncio	crebro vix	Fugit	Spectaculum
0f9e9eed-577d-47b2-9dec-ae0179c89d89	admitto adepto	vetus at	Fugit	Votum
d80aac7b-b15c-4d9c-b65c-4092bc39e41c	color conitor	sophismata vetus	Tonsor	Capto
eb99f47c-258e-4db0-ba06-2fd4a189d3ad	praesentium patria	aequus et	Cubicularis	Dens
459dd1c8-c4fc-4ee0-84c0-c7508ca0b759	admitto currus	antea decet	Tonsor	Capto
edfa04d4-2d31-4a2a-bf8f-f0d0eec58637	carbo votum	officia quidem	Utpote	Tametsi
fbde6042-9bc3d-40e3-8846-988a6a05c167	tametsi harum	tam dolorum	Utpote	Stabilis

Fuente: Elaboración propia

Ilustración 11: Respuestas de estudiantes

The screenshot shows a web application interface for an evaluation. The navigation bar at the top includes 'Home', 'Evaluations', 'Courses', 'Students', and 'Sign Out'. The user is logged in as 'TEACHER' with the email 'CDUQUE@EST.ECOTEC.EDU.EC'. The main content area is titled 'Evaluation: Fundamentos de Computación' and displays a table with columns for 'Code', 'Names', and 'Second Names'. Two rows are visible, each with a 'See answers' button. A modal window titled 'Answers from student' is open, displaying a question and its answer.

Code	Names	Second Names
0f9e4eed-577d-47b2-9dec-ae0179c89d89	admitto adeptio	vetus at
74fa9bb2-3...		tamisiium molestias

Answers from student

Question: ¿Cuál es la función principal de la CPU en un sistema informático y cómo se relaciona con la ejecución de instrucciones?

Answer: Las CPU están diseñadas para tareas generales y ejecutan instrucciones de manera secuencial, mientras que las GPU están optimizadas para realizar cálculos en paralelo, destacándose en tareas intensivas de gráficos y procesamiento masivo, como renderización 3D y aprendizaje profundo.

Pendiente de calificación

Calificar

Fuente: Elaboración propia

Módulo de Resolución de Evaluaciones

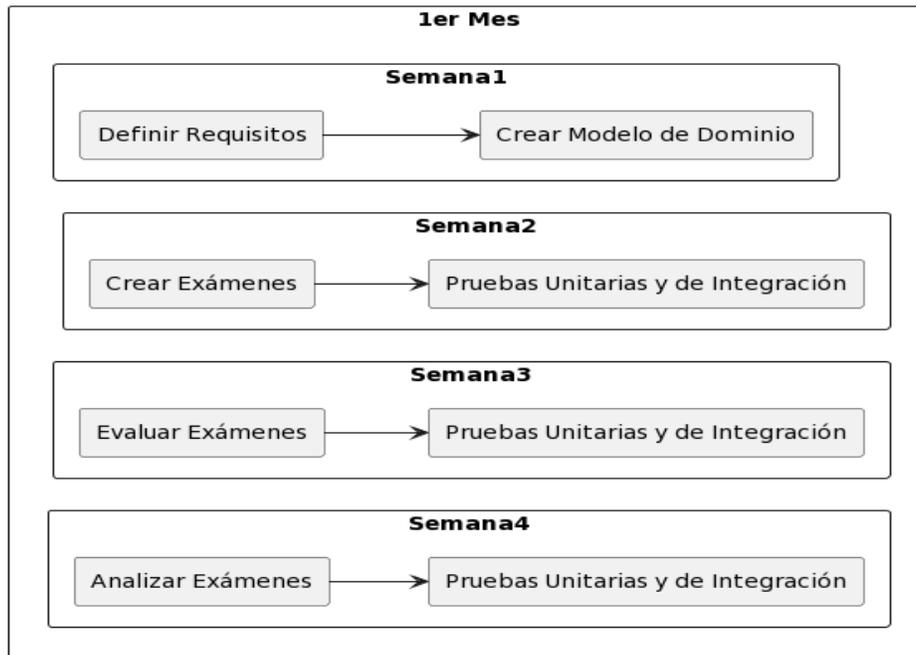
En este módulo se permite el desarrollo por parte de los estudiantes, la evaluación que se les ha asignado, permitiendo resolverla con un tiempo máximo de resolución previamente establecido por el supervisor y hasta un máximo de 1 intento para enviar sus respuestas.

La estructura del software se ha diseñado de manera intuitiva y eficiente, permitiendo una gestión fluida y completa de los cursos, evaluaciones y estudiantes involucrados en el sistema de supervisión de deshonestidad académica en entornos virtuales. Cada módulo y submódulo cumple un papel específico en la organización y administración de la información, brindando al supervisor las herramientas necesarias para llevar a cabo su labor de manera efectiva.

Cronograma del desarrollo del software

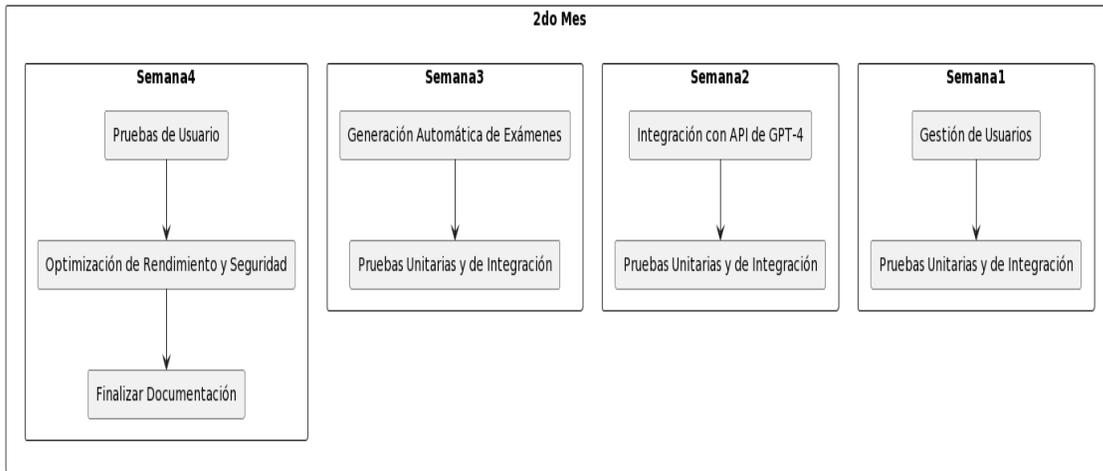
A continuación, se muestran las actividades realizadas en un periodo de 2 meses desde la fase de planificación hasta su puesta en ambiente de pruebas de la aplicación.

Ilustración 12: Cronograma de actividades parte 1



Fuente: Elaboración propia

Ilustración 13: Cronograma de actividades parte 2



Fuente: Elaboración propia

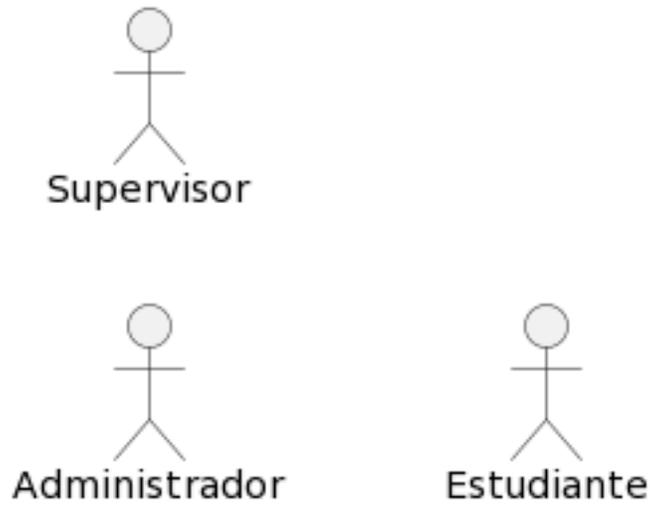
Casos de Uso

En esta sección se describirá el funcionamiento de la aplicación, basado en un sistema RBAC, un paradigma de seguridad que permite controlar las acciones que los usuarios de un sistema pueden realizar sobre sus recursos, en función de los privilegios por rol.

Determinación de actores

El sistema tiene como actores directos a administradores, supervisores y estudiantes, donde el actor principal que se centrará el análisis para cumplir con el objetivo de esta propuesta es el estudiante.

Ilustración 14: Actores de los casos de uso

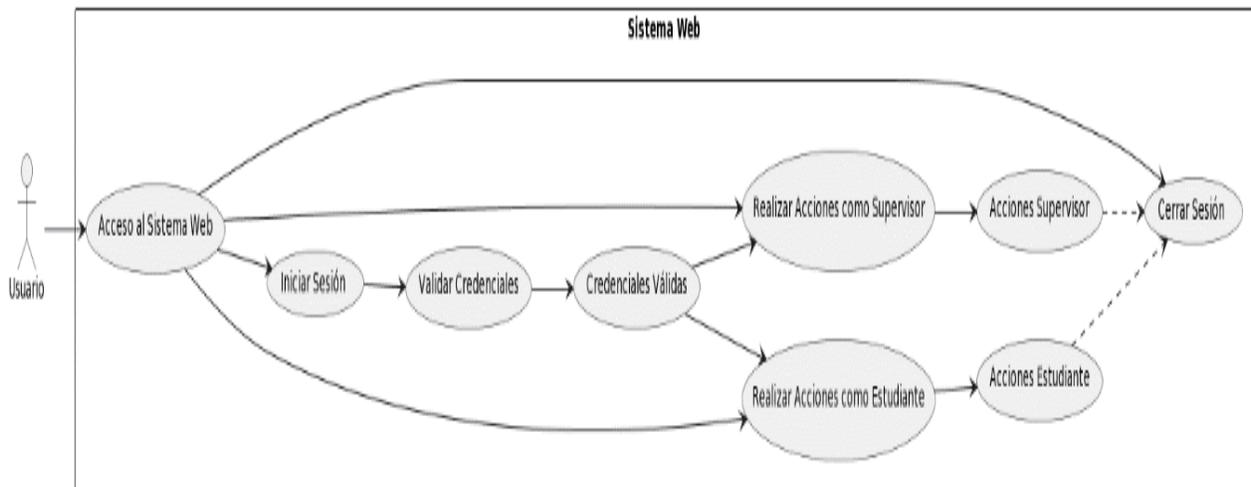


Fuente: Elaboración propia

Caso 1: Ingreso a la plataforma

En este caso se describe el proceso de ingresar al sistema a través de credenciales como usuario y contraseña, validando en la base de datos su existencia y rol asignado, para que, de esta manera, realizar acciones acordes a sus funciones permitidas.

Ilustración 15: Caso de uso 1 - Ingresar al sistema



Fuente: Elaboración propia

Caso 2: Supervisión de evaluaciones

A continuación, se presenta una descripción del proceso de consulta, creación y modificación de registros en la base de datos, para el módulo de evaluaciones. Este módulo ofrece funcionalidades clave que incluyen la creación de evaluaciones a partir de un banco de preguntas, la asignación de puntajes y cursos, y la posibilidad de iniciar la evaluación para que los usuarios con el rol de estudiante puedan completarla. Además, se proporciona la capacidad de calificar las respuestas de los estudiantes de dos formas: mediante calificación manual o utilizando el soporte de IA con GPT-4 incorporado en el sistema, lo que permite una revisión exhaustiva y precisa para asignar una calificación adecuada a cada respuesta.

Este proceso implica las siguientes etapas:

- Consulta de evaluaciones: El sistema permite realizar consultas en la base de datos para acceder a la información de evaluaciones existentes. Los usuarios autorizados pueden buscar y visualizar los registros de evaluaciones previas.
- Creación de evaluaciones: Los supervisores tienen la capacidad de crear nuevas evaluaciones. Pueden seleccionar preguntas del banco de preguntas disponible y configurar los puntajes y cursos asociados a cada evaluación.
- Modificación de evaluaciones: Los supervisores también pueden modificar los detalles de las evaluaciones existentes, como agregar o eliminar preguntas, ajustar puntajes o cambiar los cursos asignados.
- Inicio de evaluación: Una vez configurada una evaluación, el supervisor puede iniciarla para que los estudiantes puedan acceder a ella y responder las preguntas correspondientes.
- Respuestas de los estudiantes: Los estudiantes ingresan al sistema y responden las preguntas de la evaluación. El sistema registra y almacena las respuestas proporcionadas.
- Calificación de respuestas: Para calificar las respuestas de los estudiantes, el sistema brinda dos opciones. Los supervisores pueden realizar la calificación manualmente, revisando y asignando puntajes a cada respuesta. También pueden utilizar el soporte de IA con GPT-4

incorporado en el sistema, lo que les permite realizar una revisión detallada y precisa para asignar una calificación apropiada a cada respuesta.

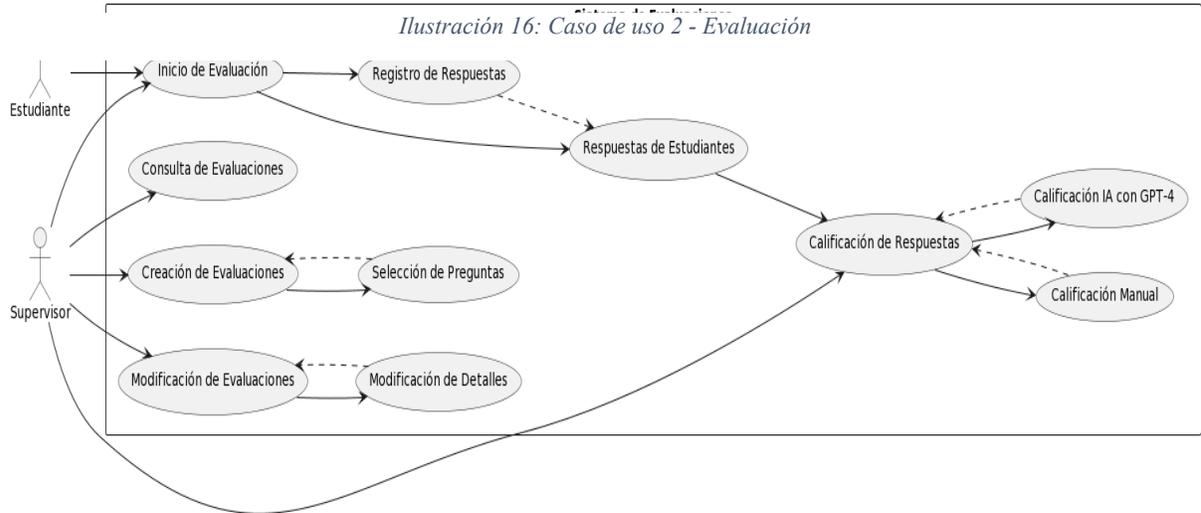


Diagrama de flujo de eventos

Fuente: Elaboración propia

En esta sección, se presenta un diagrama de flujo de eventos detallado que describe el proceso de interacción de los usuarios, ya sean supervisores o estudiantes, con el sistema web desarrollado para la supervisión de deshonestidad académica en evaluaciones teóricas en entornos virtuales. El diagrama proporciona una visión clara y secuencial de las acciones que los usuarios realizan en el sistema, desde el inicio de sesión hasta la finalización de las evaluaciones y la calificación de las respuestas. A través de este diagrama, se busca brindar una comprensión detallada del flujo de trabajo dentro del sistema, destacando las diferentes funcionalidades disponibles para cada rol y asegurando una experiencia fluida y eficiente para los usuarios.

Diagrama de interacción de objetos

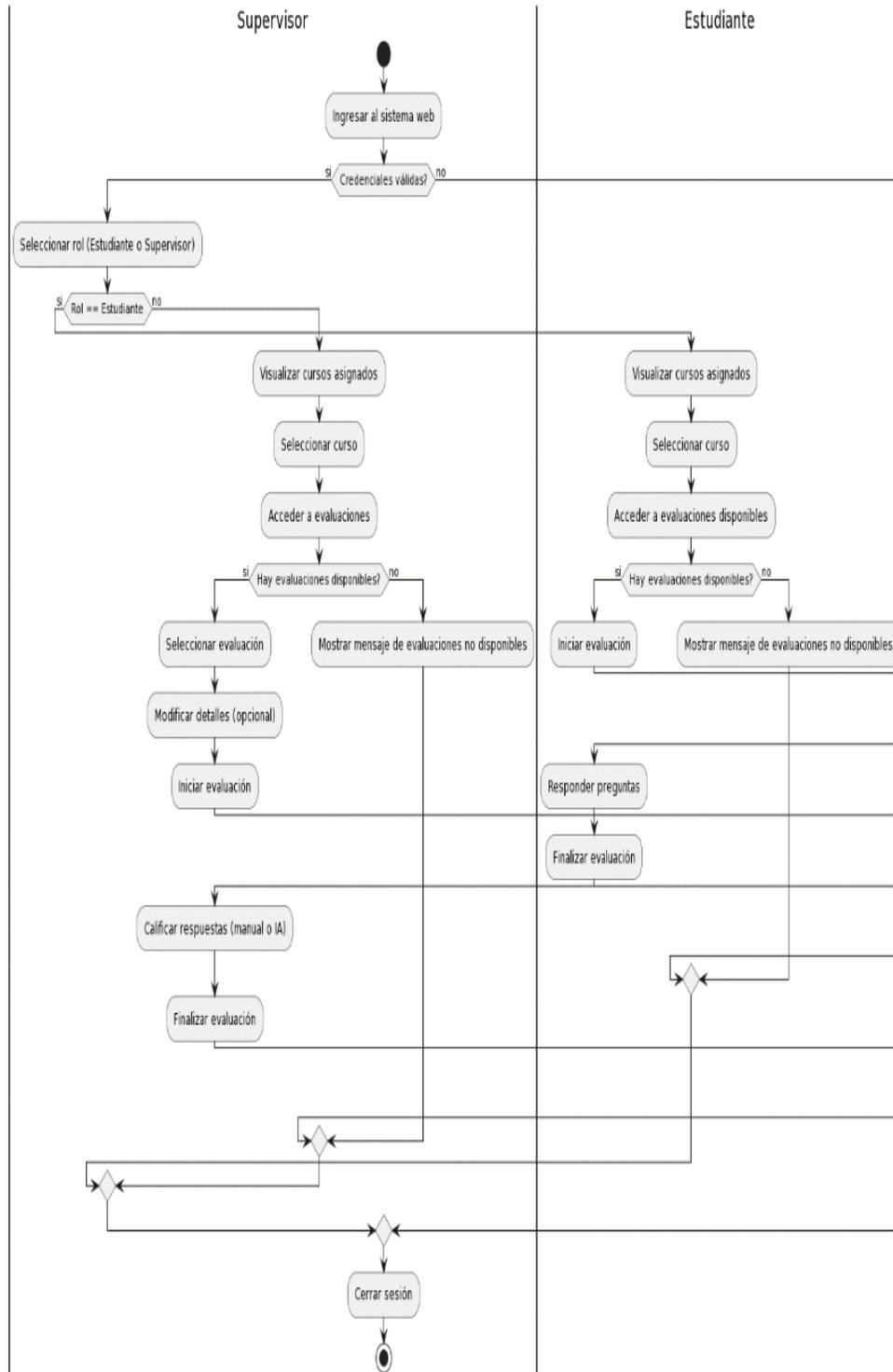
En la sección de diagrama de interacción de objetos se representa la secuencia de interacciones entre los actores y el sistema en el contexto de un sistema web utilizado en una institución educativa. El diagrama muestra las acciones realizadas por el supervisor y el estudiante, y cómo se comunican con el sistema para llevar a cabo diferentes tareas.

El supervisor inicia sesión en el sistema web ingresando sus credenciales y seleccionando su rol (supervisor o estudiante). El sistema valida las credenciales en la base de datos y devuelve una confirmación de autenticación.

Si el rol seleccionado es el de estudiante, el sistema muestra los cursos asignados al estudiante consultando la base de datos. Luego, el estudiante selecciona un curso y accede a las evaluaciones disponibles. El sistema consulta la base de datos para obtener la lista de evaluaciones y permite al estudiante iniciar una evaluación, registrando esta acción en la base de datos. El estudiante responde las preguntas y finaliza la evaluación, y el sistema registra las respuestas en la base de datos.

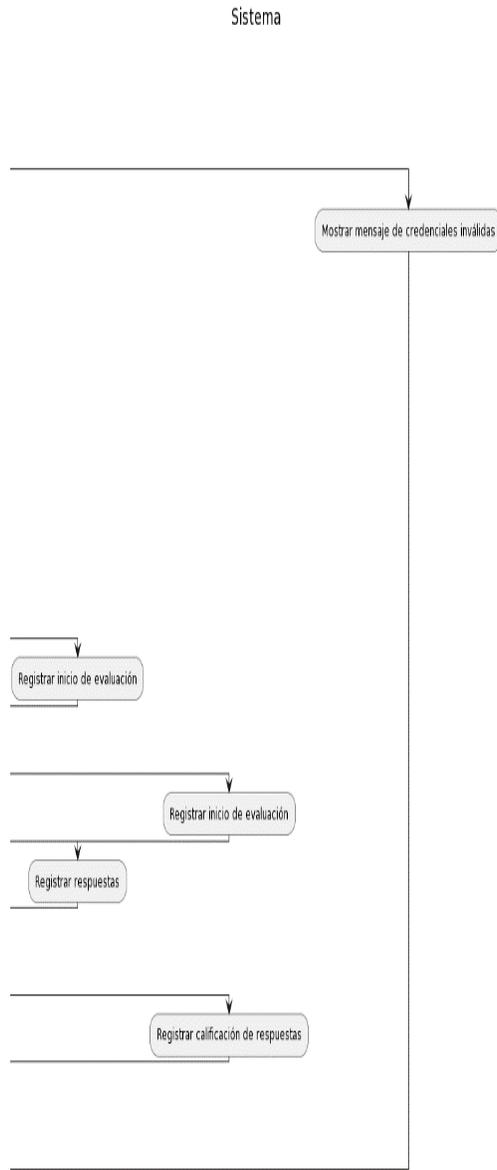
Por otro lado, si el rol seleccionado es el de supervisor, el sistema muestra los cursos asignados al supervisor consultando la base de datos. Luego, el supervisor selecciona un curso y accede a las evaluaciones disponibles. Al seleccionar una evaluación, el supervisor puede modificar los detalles de la evaluación antes de iniciarla. Posteriormente, el supervisor inicia la evaluación y califica las respuestas de los estudiantes. Al finalizar la evaluación, el sistema registra las calificaciones en la base de datos.

Ilustración 17: Diagrama de flujo de eventos parte 1



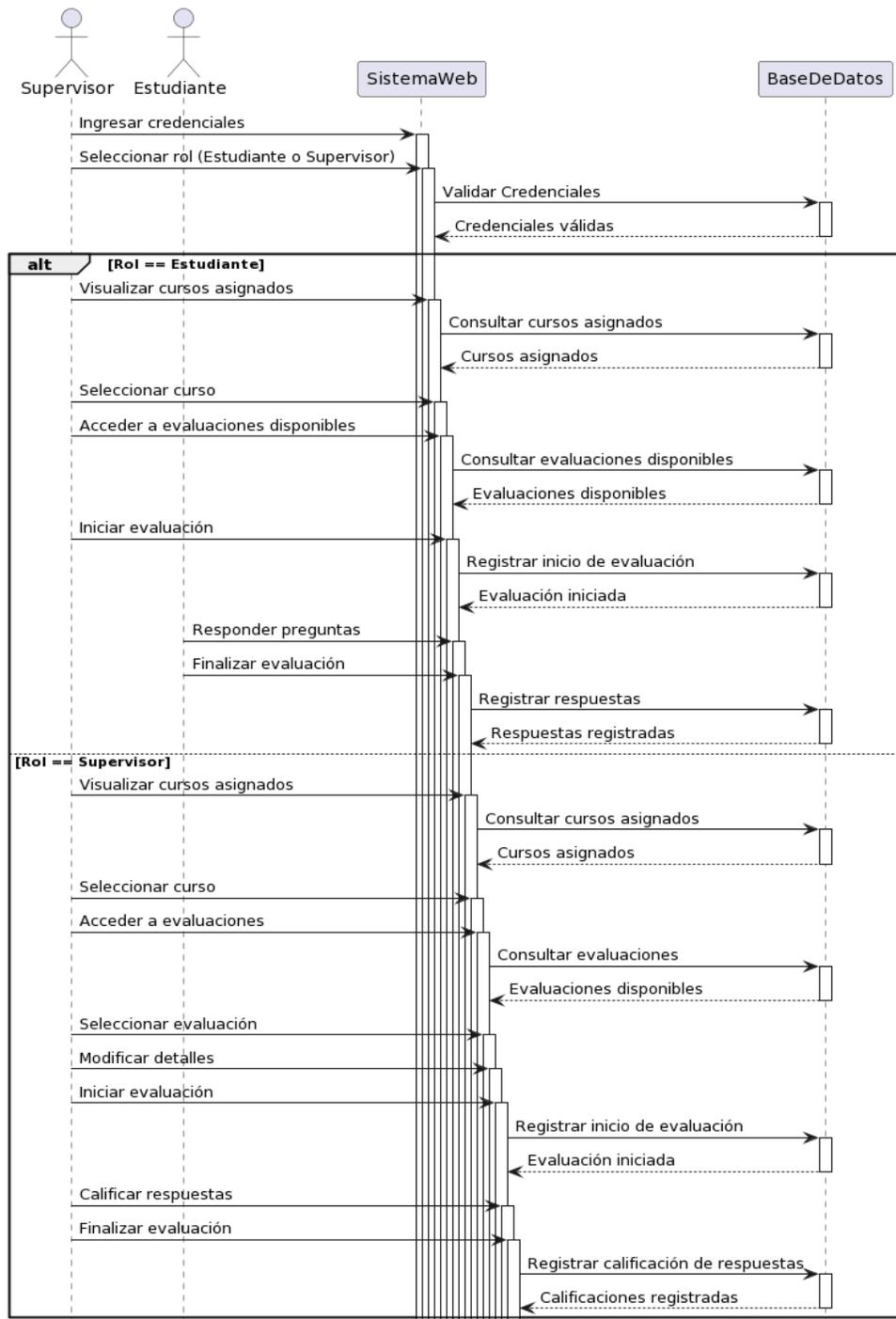
Fuente: Elaboración propia

Ilustración 18: Diagrama de flujo de eventos parte 2



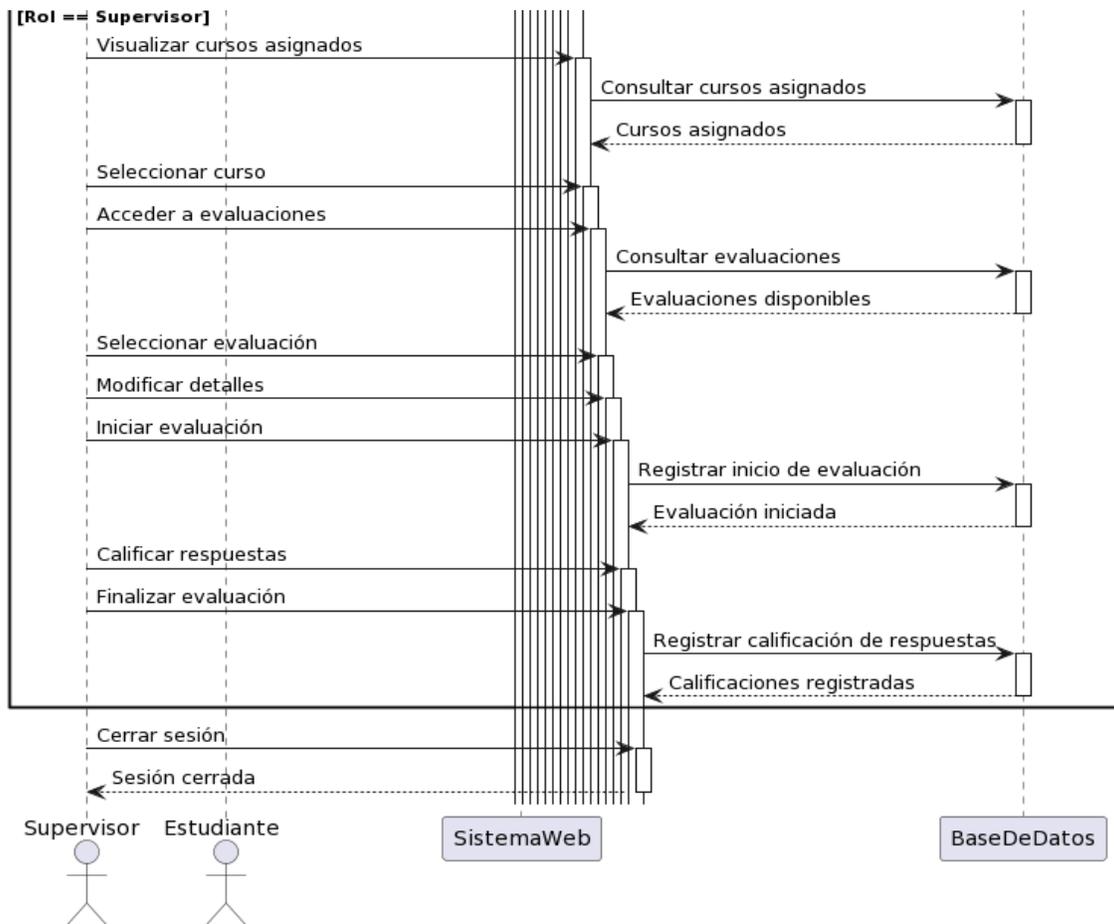
Fuente: Elaboración propia

Ilustración 19: Diagrama de interacción de objetos parte 1



Fuente: Elaboración propia

Ilustración 20: Diagrama de interacción de objetos parte 2

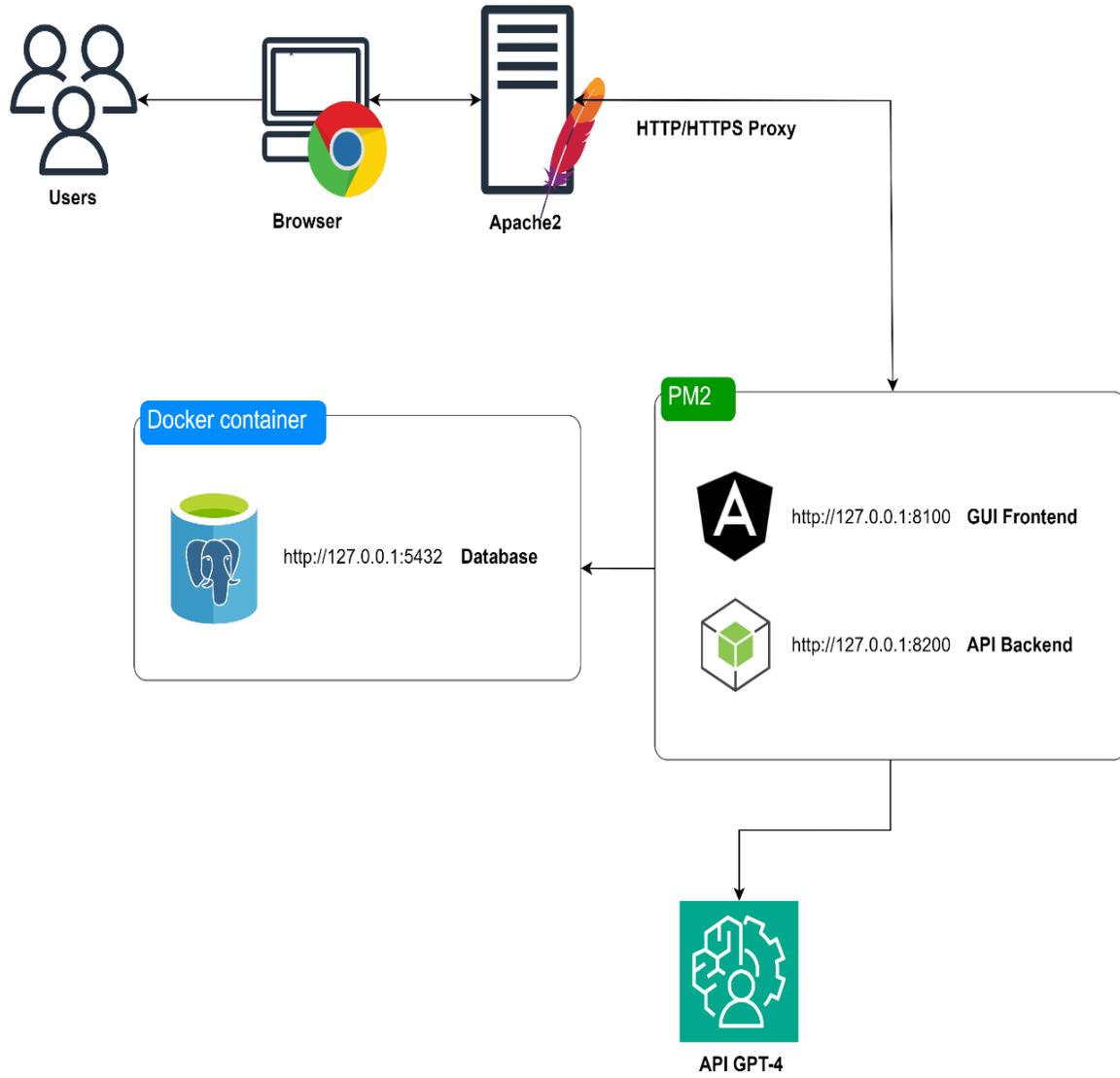


Fuente: Elaboración propia

Diagrama de arquitectura del software

A continuación, se presenta la arquitectura utilizada para el software propuesto, el cual consta de una infraestructura virtualizada que se desarrolló localmente en un entorno de sistema operativo Linux.

Ilustración 21: Arquitectura del software



Fuente: Elaboración propia

Diseño del software

En esta sección, se presenta el marco tecnológico utilizado en el desarrollo de la propuesta. Se describen las tecnologías y herramientas clave que se utilizaron para implementar el software asistido por IA para la supervisión de actos de deshonestidad académica en evaluaciones teóricas en entornos virtuales.

Librerías y entornos de ejecución

Las librerías para desarrollo web en el entorno de ejecución NodeJS utilizadas son las siguientes:

- Angular es un framework de desarrollo web utilizado para crear aplicaciones de una sola página (Single Page Applications, SPAs). Se eligió Angular debido a su capacidad para crear interfaces de usuario interactivas y su amplio ecosistema de librerías y componentes.
- Strapi es un CMS (Content Management System) de código abierto que permite la creación y gestión rápida y sencilla de contenido. Se seleccionó Strapi para la interfaz de administración de la propuesta tecnológica, facilitando la gestión de usuarios, evaluaciones y detección de deshonestidad académica.

Motor de base de datos

PostgreSQL es un sistema de gestión de bases de datos relacional y de código abierto. Se utilizó PostgreSQL como base de datos para almacenar la información relevante de usuarios, evaluaciones y registros de deshonestidad académica.

Virtualización y empaquetado de la aplicación

Docker

Docker es una plataforma de contenedores que permite el empaquetado y distribución de aplicaciones de manera independiente del entorno en el que se ejecutan. Se aprovechó Docker para crear contenedores de la propuesta tecnológica, lo cual facilitó la implementación y escalabilidad del software.

Sistema operativo

Fedora Linux

Fedora Linux es una distribución de Linux basada en software libre y de código abierto. Se eligió Fedora como sistema operativo para el despliegue de la propuesta tecnológica debido a su estabilidad y compatibilidad con las tecnologías utilizadas.

IDE de desarrollo

Visual Studio Code

Visual Studio Code es un editor de código fuente ampliamente utilizado y versátil. Se utilizó VSCode como entorno de desarrollo integrado (IDE) para la escritura y depuración del código de la propuesta tecnológica, aprovechando sus numerosas extensiones y herramientas para el desarrollo web.

API de comunicación con Inteligencia Artificial

GPT-4

GPT-4 API es un potente modelo de lenguaje de gran tamaño (LLM) desarrollado por OpenAI que puede ayudar en una gran variedad de tareas, desde escribir correos electrónicos hasta generar código. Este modelo supone una mejora significativa con respecto a los anteriores modelos de IA generativa de OpenAI, sobre todo en la forma en que gestiona las indicaciones complejas y matizadas.

Conclusiones

Se concluye que los objetivos de investigación presentados son muy pertinentes para abordar el problema de la deshonestidad académica en los entornos académicos en línea. Al identificar las

formas predominantes de deshonestidad, diseñar un prototipo de software apoyado en inteligencia artificial, ejecutar pruebas funcionales y evaluar la eficacia del software, este proyecto tiene el potencial de mejorar significativamente la integridad y la seguridad de los exámenes en línea en las instituciones de educación superior.

Recomendaciones

Para mejorar aún más la eficacia del software de supervisión de exámenes en entornos académicos en línea, recomendamos integrar capacidades avanzadas de detección de plagio. Mientras que el sistema actual se centra en la comparación de las respuestas de los estudiantes a las pruebas generadas dentro del sistema, la incorporación de algoritmos apoyados por IA para la detección de plagio proporcionaría una evaluación exhaustiva de la integridad académica. Esta funcionalidad permitiría al software comparar las respuestas de los estudiantes no sólo dentro del sistema, sino también con una amplia base de datos de recursos académicos y trabajos presentados anteriormente. De este modo, el programa informático puede detectar posibles casos de plagio y proporcionar a los educadores información valiosa para abordar eficazmente la deshonestidad académica.

Con el fin de proporcionar a los educadores una comprensión más completa del rendimiento de los estudiantes y de los posibles casos de deshonestidad académica, se recomienda ampliar las funciones de análisis de datos y elaboración de informes del software de supervisión de exámenes. Aprovechando las capacidades de inteligencia artificial y la API de GPT, el software puede generar informes gráficos detallados que destaquen patrones y anomalías en las respuestas de los estudiantes. Esto permitiría a los educadores identificar fácilmente posibles comportamientos fraudulentos y tomar las medidas adecuadas. Además, el software debería proporcionar un análisis comparativo de las respuestas individuales de los estudiantes a las pruebas generadas, lo que permitiría a los educadores identificar la posible colaboración o el intercambio no autorizado de respuestas. Al ampliar las funciones de análisis de datos y elaboración de informes, el software

puede proporcionar a los educadores información valiosa para mejorar la integridad y la seguridad de los exámenes en línea.

Bibliografía

Abdelaal, Elsayed & Walpita Gamage, Sithara & Mills, Julie. (2019). Artificial Intelligence Is a Tool for Cheating Academic Integrity.

Academic Dishonesty Definition and Types | Academic Integrity Tutorial | Northern Illinois University. (n.d.). Northern Illinois University. <https://www.niu.edu/academic-integrity/faculty/types/index.shtml>

Academic Dishonesty, Definitions and Examples. (2017, December 11). John Jay College of Criminal Justice. <https://jjay.cuny.edu/academic-dishonesty-definitions-and-examples>

Academic Dishonesty: Reasons, Consequences And Preventions. (2021, September 23). Edubirdie. Retrieved July 12, 2023, from <https://edubirdie.com/examples/academic-dishonesty-reasons-consequences-and-preventions/>.

Academic misconduct in higher education: A comprehensive review. *Journal of Higher Education Policy And Leadership Studies*, 1(1), 25-45. <https://doi.org/10.29252/johepal.1.1.25>

Alsabhan, W. (2023, April 20). Student Cheating Detection in Higher Education by Implementing Machine Learning and LSTM Techniques. *Sensors*; MDPI. <https://doi.org/10.3390/s23084149>

Bhandari, P. (2023, June 22). What is a Likert scale? | Guide & Examples. Scribbr. <https://www.scribbr.com/methodology/likert-scale/>

Bretag, T., Harper, R., Burton, M., Ellis, C., Newton, P., van Haeringen, K., Saddiqui, S., & Rozenberg, P. (2020). Contract cheating: a survey of Australian university students. *Studies in Higher Education*, 45(11), 2174-2186. <https://doi.org/10.1080/03075079.2018.1462788>

Bushwick, S. (2023, March 16). What the new GPT-4 AI can do. *Scientific American*. <https://www.scientificamerican.com/article/what-the-new-gpt-4-ai-can-do>

Chiang, F., Zhu, D., & Yu, W. (2022). A systematic review of academic dishonesty in online learning environments. *Journal of Computer Assisted Learning*, 38(4), 907–928. <https://doi.org/10.1111/jcal.12656>

Chiang, F., Zhu, D., & Yu, W. (2022). A systematic review of academic dishonesty in online learning environments. *Journal of Computer Assisted Learning*, 38(4), 907–928.

<https://doi.org/10.1111/jcal.12656>

Cotton, D., Cotton, P., & Shipway, J. R. (2023). Chatting and cheating. Ensuring academic integrity in the era of ChatGPT. <https://doi.org/10.35542/osf.io/mrz8h>

Daniel, S. J. (2020). Education and the COVID-19 pandemic. *Prospects*, 49(1), 91–96. <https://doi.org/10.1007/s11125-020-09464-3>

Holden, O. L., Norris, M. E., & Kuhlmeier, V. A. (2021). Academic Integrity in Online Assessment: A Research review. *Frontiers in Education*, 6.

<https://doi.org/10.3389/feduc.2021.639814>

Holden, O. L., Norris, M. E., & Kuhlmeier, V. A. (2021). Academic Integrity in Online Assessment: A Research review. *Frontiers in Education*, 6.

<https://doi.org/10.3389/feduc.2021.639814>

Is AI a threat to academic integrity? (2022, January 13). Ouriginal. <https://www.ouriginal.com/is-ai-a-threat-to-academic-integrity/>

Knox, L. (2023, April 3). Turnitin’s solution to AI cheating raises faculty concerns. *Inside Higher Ed | Higher Education News, Events and Jobs*.

<https://www.insidehighered.com/news/2023/04/03/turnitins-solution-ai-cheating-raises-faculty-concerns>

Kumar, R., Eaton, S. E., Mindzak, M., & Morrison, R. (2023). Academic integrity and artificial intelligence: An overview. *Handbook of Academic Integrity*, 1-14.

https://doi.org/10.1007/978-981-287-079-7_153-1

Malik, A. A., Hassan, M., Rizwan, M., Mushtaque, I., Lak, T. A., & Hussain, M. (2023). Impact of academic cheating and perceived online learning effectiveness on academic performance

during the COVID-19 pandemic among Pakistani students. *Frontiers in Psychology*, 14.

<https://doi.org/10.3389/fpsyg.2023.1124095>

Okoye, K., Hussein, H., Arrona-Palacios, A., Quintero, H. N., Ortega, L. O., Sanchez, A. L., Ortiz, E. A., Escamilla, J., & Hosseini, S. (2022). Impact of digital technologies upon teaching and learning in higher education in Latin America: An outlook on the reach, barriers, and bottlenecks. *Education and Information Technologies*, 28(2), 2291-2360. <https://doi.org/10.1007/s10639-022-11214-1>

Parnther, C. (2020). Academic misconduct in higher education: A comprehensive review. *Journal of Higher Education Policy And Leadership Studies*, 1(1), 25-45.

<https://doi.org/10.29252/johepal.1.1.25>

Quiz #2. (n.d.). <https://www.rochester.edu/college/honesty/policy/quiz2.html>

Surahman, E., & Wang, T. (2022). Academic dishonesty and trustworthy assessment in online learning: A systematic literature review. *Journal of Computer Assisted Learning*, 38(6), 1535-1553. <https://doi.org/10.1111/jcal.12708>

Toapanta, S. M., Díaz, E. Z., Vizuete, O. M., & Chávez, E. E. (2022). Analysis of artificial intelligence applied in virtual learning environments in higher education for Ecuador. *Frontiers in Artificial Intelligence and Applications*. <https://doi.org/10.3233/faia220563>

Ventayen, R. J. (2023). ChatGPT by OpenAI: Students' viewpoint on cheating using artificial intelligence-based application. *SSRN Electronic Journal*.

<https://doi.org/10.2139/ssrn.4361548>

What is GPT-4 and Why Does it Matter? (2023). Learn R, Python & Data Science Online | DataCamp. <https://www.datacamp.com/blog/what-we-know-gpt>

Zawacki-Richter, O., Marín, V. I., Bond, M., & Gouverneur, F. (2019). Systematic review of research on artificial intelligence applications in higher education – where are the educators?

International Journal of Educational Technology in Higher Education, 16(1).

<https://doi.org/10.1186/s41239-019-0171-0>