



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL
FACULTAD DE INGENIERÍA
CARRERA DE COMPUTACIÓN**

TEMA:

Prototipo de tutor inteligente para la gestión autónoma de tutorías académicas en la asignatura de programación orientada a objetos en la carrera de computación de la Universidad Católica de Santiago de Guayaquil.

AUTOR:

Páliz Sánchez, Rodolfo David

Proyecto de tecnología de información previo a la obtención del título de INGENIERO EN CIENCIAS DE LA COMPUTACIÓN

TUTOR:

Castro Aguilar, Gilberto Fernando

**Guayaquil, Ecuador
4 de marzo del año 2026**



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL
FACULTAD DE INGENIERÍA
CARRERA DE COMPUTACIÓN**

CERTIFICACIÓN

Certificamos que el presente proyecto de tecnología de información fue realizado en su totalidad por **Páliz Sánchez, Rodolfo David**, como requerimiento para la obtención del título de **Ingeniero en Ciencias de la Computación**.

TUTOR

f. _____

Castro Aguilar, Gilberto Fernando

Guayaquil, a los 04 días del mes de marzo del año 2026.



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL
FACULTAD DE INGENIERÍA
CARRERA DE COMPUTACIÓN**

DECLARACIÓN DE RESPONSABILIDAD

Yo, Páliz Sánchez, Rodolfo David

DECLARO QUE:

El proyecto de tecnología de información, **Prototipo de tutor inteligente para la gestión autónoma de tutorías académicas en la asignatura de programación orientada a objetos en la carrera de computación de la Universidad Católica de Santiago de Guayaquil**, previo a la obtención del título de **Ingeniero en Ciencias de la Computación**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias bibliográficas. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 04 días del mes de marzo del año 2026

EL AUTOR

f. _____

Páliz Sánchez, Rodolfo David



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL
FACULTAD DE INGENIERÍA
CARRERA DE COMPUTACIÓN

AUTORIZACIÓN

Yo, **Páliz Sánchez, Rodolfo David**

Autorizo a la Universidad Católica de Santiago de Guayaquil a la **publicación** en la biblioteca de la institución del proyecto de tecnología de información, **Prototipo de tutor inteligente para la gestión autónoma de tutorías académicas en la asignatura de programación orientada a objetos en la carrera de computación de la Universidad Católica de Santiago de Guayaquil**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 04 días del mes de marzo del año 2026

EL AUTOR:

f. _____

Páliz Sánchez, Rodolfo David



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL
FACULTAD DE INGENIERÍA
CARRERA DE COMPUTACIÓN

REPORTE ANTIPLAGIO



CERTIFICADO DE ANÁLISIS
magister

final-Correccion numero 1 APLICADA
Paliz_Trabajo de Titulación

4%
Textos
sospechosos

Nombre del documento: final-Correccion numero 1 APLICADA
Paliz_Trabajo de Titulación.docx
ID del documento: 0acd6c7712ed6a148370686c1c89538b57523e4d
Tamaño del documento original: 3,62 MB

Depositante: Gilberto Fernando Castro Aguilar
Fecha de depósito: 19/2/2026
Tipo de carga: interface
fecha de fin de análisis: 19/2/2026

Númer
Númer

TUTOR

f. _____

Castro Aguilar, Gilberto Fernando

AGRADECIMIENTO

Desde lo más profundo de mi corazón y con mucha humildad, agradezco a Dios por ser mi guía en cada uno de los pasos de este camino, por darme la fortaleza en los momentos difíciles y la sabiduría para no rendirme ante todos los obstáculos que me pusieron en frente. A mi madre, Anany, por su amor infinito, por sus sacrificios silenciosos y por creer en mí incluso cuando yo mismo quería rendirme; a mi esposa, Jennifer, por su paciencia, su apoyo sin condición, su amabilidad, su amor y por ser mi refugio en los momentos de cansancio; y a mi hijo, Yusei, quien con su sola existencia se convirtió en mi mayor inspiración y razón para seguir adelante, recordándome cada día el verdadero sentido de esta lucha y penitencia.

De manera muy especial, expreso mi más sincero agradecimiento a mi profesor Edison Toala, por su dedicación, su guía constante y por compartir no solo sus conocimientos, sino también su motivación y confianza en mi capacidad para alcanzar esta meta. Asimismo, agradezco de corazón a mi grupo de amigos: Rex, Turbo, Mi lidel, El patrón, Raw, Neymar, Mole, Liz, Bolita, Jairo deidad, Mr. Arturo, Mr. Víctor, Blanquita, Jully, Randy, Blade y Mi Fofa, quienes, con su apoyo, su aliento y su compañía hicieron este proceso más llevadero y significativo. Finalmente, agradezco a mis compañeros de universidad, con quienes compartí desafíos, aprendizajes y momentos inolvidables; aunque no los mencione individualmente, cada uno ha sido parte fundamental de este logro que, a pesar de las dificultades, los retos y las piedras en el camino, hoy culmino de pie, con orgullo y gratitud.

DEDICATORIA

Dedico este trabajo a quienes ya no están físicamente a mi lado, pero cuya ausencia se ha convertido en la presencia más profunda en mi vida. Mi padre, Pedro, y a mi madre, Inés, quienes fueron mi origen, mi guía y mi mayor ejemplo de amor y fortaleza. Este logro lleva sus nombres en el silencio, porque todo lo que yo soy y todo lo que hoy alcanzo nace de ustedes; su fuego, su lucha, su voluntad y todo lo que ustedes representan sigue vivo en mí, seguirá vivo en Yusei y en todo lo que hagamos las próximas generaciones de esta familia que ustedes formaron. Así mismo seguirán vivos en nuestras memorias. Aunque su partida dejó un vacío imposible de llenar, su amor sigue siendo la luz que nos sostiene y nos impulsa a seguir adelante.

A mis queridos compañeros animales Toby, Accel y mi gatita Kitty, almas nobles e inocentes que me enseñaron el estado más puro y primigenio del amor, la lealtad y la compañía incondicional. Su ausencia duele, pero su recuerdo vive en cada paso que doy, en cada momento de alegría y en cada logro alcanzado. Esta meta y todas las que logre alcanzar serán de ustedes porque de una u otra forma, caminaron conmigo hasta aquí. No se han ido del todo, nunca se irán del todo, porque viven eternamente en mi corazón.



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL
FACULTAD DE INGENIERÍA
CARRERA DE COMPUTACIÓN**

TRIBUNAL DE SUSTENTACIÓN

f. _____

ING. ANA CAMACHO CORONEL, MGS
DIRECTORA DE CARRERA

f. _____

ING. ROBERTO GARCIA SÁNCHEZ, PhD
COORDINADOR DEL ÁREA

f. _____

ING. JOSÉ ERAZO AYON, MGS
OPONENTE

ÍNDICE

Resumen	XIII
Abstract	XIV
Introducción	2
Capítulo I	4
El Problema	4
ANÁLISIS DEL PROBLEMA.....	4
Ubicación del Problema en un Contexto	4
Causas y Consecuencias del Problema	5
Delimitación del Problema.....	6
Formulación del Problema.....	6
Evaluación del Problema.....	7
OBJETIVOS.....	9
Objetivo General	9
Objetivos Específicos	9
ALCANCES DEL PROBLEMA.....	9
JUSTIFICACION E IMPORTANCIA.....	10
PREGUNTA DE INVESTIGACIÓN	11
VARIABLES DE LA INVESTIGACIÓN.....	11
Variables Independientes: Python, (Backend),React (Frontend), Azure OpenAi (motor de IA), ProgrestSQL (Estancia).....	12
Variable Dependiente 1: Prototipo de Tutor Inteligente.	12
Capítulo II	13
Marco Teórico	13
Antecedentes del estudio	13
Etimología y Concepto.....	13
Herramientas de Desarrollo	16
Lenguajes y tecnologías de desarrollo.....	16
HTML	16
CSS.....	16
Tecnologías de desarrollo backend	16
Python.....	17
FastAPI	17
Base de datos.....	17
PostgreSQL.....	17
Infraestructura en la nube	18
Microsoft Azure	19
Integración de inteligencia artificial	19
Control de versiones	20
Git.....	20
Capítulo III	21
Metodología De La Investigación	21
Tipo de Investigación.....	22

Investigación Descriptiva.....	22
Investigación Aplicada.....	22
Enfoque de la Investigación.....	22
Población y Muestra	23
Población	23
Muestra	23
Técnicas e Instrumentos de Recolección de Datos	23
Técnica:	24
Instrumento:.....	24
ANÁLISIS E INTERPRETACIÓN DE RESULTADOS.....	24
Conclusiones del análisis.....	29
Metodología de Desarrollo de Software.....	29
Capítulo IV	31
Propuesta Tecnológica	31
HERRAMIENTAS DE DESARROLLO	31
Estructura de la base de datos.	31
Modelos usados con el programa de estudiantes de Azure.	33
Pantalla de inicio.....	34
Inicio: selección de participantes:	34
Dashboard de profesores	35
Descripción del panel de gestión y visualización de tutorías	35
Conclusiones	36
Recomendaciones	38
Referencias Bibliográficas	39

INDICE DE IMÁGENES

ILUSTRACIÓN 1 ÁRBOL DE PROBLEMAS	6
ILUSTRACIÓN 2 SISTEMA TUTOR INTELIGENTE	15
ILUSTRACIÓN 3 DISTRIBUCIÓN DE ESTUDIANTES POR SEMESTRE	25
ILUSTRACIÓN 4 NIVEL DE CONOCIMIENTO AUTO PERCIBIDO EN POO	26
ILUSTRACIÓN 5 PERCEPCIÓN DE DIFICULTAD DE LA ASIGNATURA	27
ILUSTRACIÓN 6 NECESIDAD DE TUTORÍAS ACADÉMICAS	27
ILUSTRACIÓN 7 VALIDACIÓN DE LA PROPUESTA TECNOLÓGICA	28
ILUSTRACIÓN 8 DISPOSICIÓN DE USO DE LA HERRAMIENTA	29
ILUSTRACIÓN 9 ESTRUCTURA DE LA BASE DE DATOS.	31
ILUSTRACIÓN 10 SCRIPT DE INGESTA DE DATOS.	32
ILUSTRACIÓN 11 IMPLEMENTACIÓN DE MODELO.	33
ILUSTRACIÓN 12 PANTALLA DE INICIO	34
ILUSTRACIÓN 13 FUNCIONALIDADES	34
ILUSTRACIÓN 14 DASHBOARD DE PROFESORES.....	35
ILUSTRACIÓN 15 DESCRIPCIÓN DEL PANEL DE GESTIÓN Y VISUALIZACIÓN DE TUTORÍAS ..	35
ILUSTRACIÓN 16 ENCUESTA DE DISTRIBUCIÓN DE LA POBLACIÓN ESTUDIANTIL POR SEMESTRE.....	40
ILUSTRACIÓN 17 ENCUESTA AUTOPERCEPCIÓN DEL NIVEL DE CONOCIMIENTO EN PROGRAMACIÓN ORIENTADA A OBJETOS.	40
ILUSTRACIÓN 18 ENCUESTA SOBRE EL USO DE TECNOLOGÍAS Y APRENDIZAJE AUTÓNOMO.	41
ILUSTRACIÓN 19 ENCUESTA DEMANDA Y NECESIDAD DE TUTORÍAS ACADÉMICAS.	41

ÍNDICE DE TABLAS

TABLA 1 CRITERIOS PARA LA DELIMITACIÓN DEL PROBLEMA	6
TABLA 2 EVALUACIÓN DEL PROBLEMA	7
TABLA 3 CRITERIOS COMPARATIVOS DE BASE DE DATOS RELACIONALES	18
TABLA 4 COMPARACIÓN DE PLATAFORMAS DE COMPUTACIÓN EN LA NUBE	19
TABLA 5 POBLACIÓN DE ESTUDIO	23
TABLA 6 MUESTRA DEL ESTUDIO	23

Resumen

Durante la carrera de Ingeniería en ciencias de la computación y sus respectivas asignaturas como las tutorías académicas son un elemento fundamental que además de servir como una calificación en el porcentaje académico de aprobación sirve enormemente al aporte de conocimiento y aprendizaje de los estudiantes; agregado a esto la materia de programación orientada a objetos (POO) es un eje primordial en la formación de los estudiantes, sin embargo, el nivel de abstracción y complejidad en cada uno de sus conceptos que posee la materia provoca la generación de aquellas dificultades continuas en el aprendizaje de los estudiantes. En el presente trabajo de integración curricular se propone el desarrollo de una herramienta interactiva que tiene como finalidad poder fortalecer la autonomía de los estudiantes, facilitar la comprensión a conceptos referentes de la materia de POO y la consolidación de información relevante de la asignatura.

Palabras Clave: *Programación Orientada a Objetos, abstracción, aprendizaje, herramienta didáctica.*

Abstract

During the computer science engineering degree program, academic tutorials are a fundamental element that, in addition to serving as a grade in the academic pass rate, greatly contribute to students' knowledge and learning. In addition to this, the subject of OOP (object-oriented programming) is a fundamental part of the students' education. However, the level of abstraction and complexity in each of the concepts covered in the subject causes continuous difficulties in the students' learning. This curriculum integration project proposes the development of an interactive tool designed to strengthen student autonomy, facilitate understanding of concepts related to OOP (object-oriented programming), and consolidate relevant information about the subject.

Key words: *Object-Oriented Programming, abstraction, learning, teaching tool.*

Introducción

El proceso enseñanza- aprendizaje no es algo que se limite únicamente a la labor de asistir a un aula en un horario determinado de la semana, por el contrario, es una actividad que se realiza en todo momento y que se ve afectada por factores como el tiempo, las costumbres las relaciones intrapersonales e interpersonales (menciónese amigos, compañeros, familiares etc.), el estilo de aprendizaje en cada uno, entre otros. Agregado a esto la formación en el área de la computación exige a los estudiantes que desarrollen habilidades tanto analíticas como técnicas que les permitan comprender y aplicar conceptos fundamentales del desarrollo y el entendimiento de un software.

En este proceso la asignatura de programación orientada a objetos ocupa un lugar céntrico, al proporcionar los principios necesarios para la modelación, estructuración y el mantenimiento de sistemas informáticos eficientes. No obstante, la adquisición y practica de estos conceptos no siempre resulta inmediata, esto a causa del nivel de abstracción que implican, ejemplos más comunes son: el encapsulamiento, herencia, polimorfismo entre otros, los cuales suelen representar un punto de inflexión en la trayectoria académica del estudiante.

En este contexto, el uso de tecnologías inteligentes que van aplicadas a la educación superior abre nuevas posibilidades para mejorar los procesos de tutorías académicas. Ante estas dificultades, las tutorías académicas se convierten en un recurso relevante para reforzar el aprendizaje y atender necesidades específicas que no siempre pueden ser abordadas en el aula.

El principal propósito de este trabajo de integración curricular es Diseñar e Implementar una interfaz web que le permita a los estudiantes obtener una tutoría autónoma con respecto a los temas relacionados a POO ligada al silabo de la

materia complementada con los temas más importantes de la misma y registrarla con el fin de obtener una prueba de las tutorías empleadas por los docentes a los estudiantes.

La implementación de este prototipo de tutor inteligente contribuirá al aprendizaje y mejor comprensión de los temas más importantes de la materia de POO de los estudiantes de Ingeniería en ciencias de la computación de la Universidad Católica de Santiago de Guayaquil y al registro que evidencie el trabajo presentado por los docentes.

Mediante este Prototipo de tutor inteligente se ofrece un entorno de instrucción que se adapta a los requerimientos y necesidades de aprendizaje del estudiante que a su vez complementa y refuerza aún más la instrucción profesor-alumno. Cabe mencionar también que el uso de tecnologías aplicadas a la educación superior abre nuevas posibilidades para mejorar los procesos de tutorías. Es por eso por lo que para la situación que acontece es necesario replantear los modelos tradicionales de apoyo académico en especial en entornos educativos donde la demanda y la diversidad de niveles de conocimiento son elevadas.

El mayor y gran importante objetivo del presente trabajo de integración curricular es ser una ayuda complementaria y un indicador de lo útil que son las herramientas inteligentes, su uso y exploración como apoyo al proceso de tutorías académicas en la asignatura de POO considerando las necesidades de los estudiantes respecto a temas complicados que habitan en la asignatura. Sin olvidar además que le será de ayuda a los docentes por el registro autónomo de las búsquedas realizadas por estudiantes que evidenciaran el trabajo hecho por ellos.

Capítulo I

El Problema

Descripción del problema

En el proceso de enseñanza y aprendizaje de la asignatura de POO de la Universidad Católica de Santiago de Guayaquil el estudiante cuenta con interacciones con el docente a nivel presencial, realizando actividades en conjunto con un seguimiento más cercano debido al acto presencial, adicionando el estudiante puede recurrir a ayudantías de cátedra en el caso de requerirlas.

No obstante, la complejidad que poseen los contenidos de la asignatura, sumada al nivel de abstracción requerido, suele generar problemas en su comprensión y aplicación práctica por parte de los estudiantes.

Cabe recalcar que el estudiante no tiene un recurso pedagógico diferente a los ya mencionados por lo que se propone crear un tipo de tutor inteligente con el propósito de facilitar el acceso a más conocimiento a través de búsquedas semánticas ligadas estrictamente al sílabo y a los temas más importantes de la asignatura sumado al registro inteligente de dichas tutorías para la evidencia del trabajo realizado por los docentes.

ANÁLISIS DEL PROBLEMA.

Ubicación del Problema en un Contexto

El problema se sitúa en la Universidad Católica de Santiago de Guayaquil, dentro de la Carrera de Computación. Donde se proporciona el proceso de enseñanza-aprendizaje de la asignatura de Programación Orientada a Objetos

(POO) de igual forma en aquellos procesos administrativos y pedagógicos vinculados a la gestión y seguimiento de las tutorías académicas.

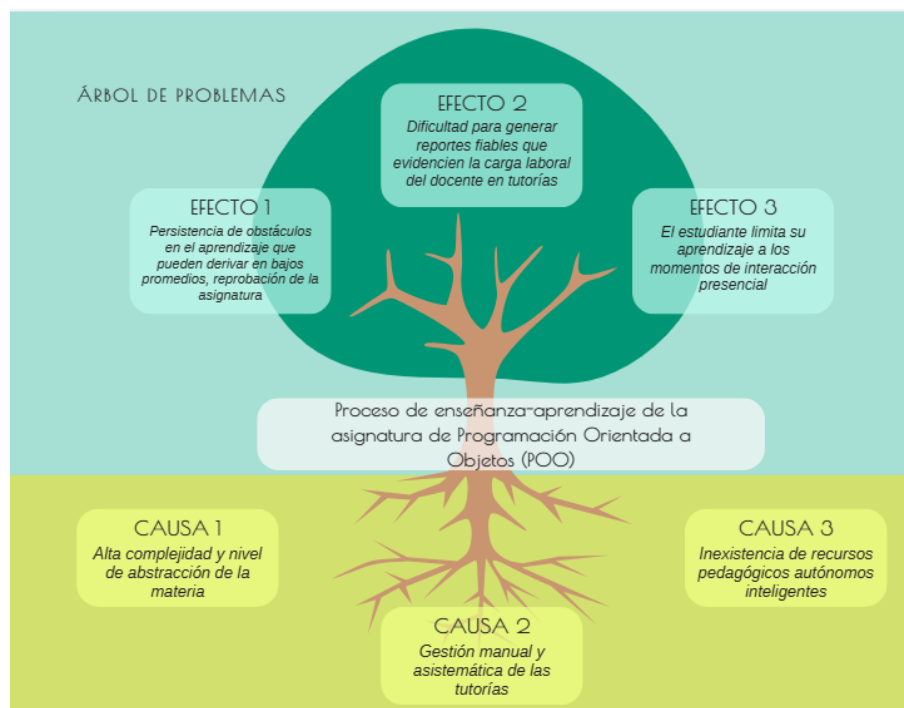
Causas y Consecuencias del Problema

Para poder entender la problemática que afecta en el proceso de enseñanza-aprendizaje de la asignatura de Programación Orientada a Objetos (POO) y en la gestión de tutorías académicas, se llevó a cabo un análisis de relación causa-efecto.

Desde este análisis se identificaron factores claves para esta problemática que son tanto pedagógicos al igual que tecnológicos tal como la falta de herramientas de apoyo autónomo, y la dificultad en la selección de conceptos claves para la asignatura. Por otro lado, se observó que en la parte administrativa también es afectada al ejecutar procesos manuales y asistemáticos, lo cual impiden un monitoreo eficiente y limita la generación de evidencias sobre el rendimiento de los estudiantes y del docente en la asignatura.

A continuación, se presenta el esquema del Árbol de Problemas, donde se detalla de forma estructurada la relación entre las causas, el problema central y sus efectos.

Ilustración 1 Árbol de problemas



Nota: Representación esquemática de los factores que obstaculizan el aprendizaje.

Delimitación del Problema

La delimitación del problema se definió con los siguientes 4 criterios, los cuales se pueden observar en la siguiente tabla:

#	Criterios	Detalles
1	Campo	Ciencias de la Computación
2	Área	Ingeniería de Software y Sistemas Inteligentes.
3	Aspecto	Automatización de procesos académicos y asistencia pedagógica.
4	Objeto de Estudio	Proceso de gestión de tutorías y el aprendizaje de conceptos claves de POO.

Formulación del Problema

¿De qué manera el desarrollo de un prototipo de tutor inteligente podrá influir en la optimización de la gestión académica y en el fortalecimiento del aprendizaje de los conceptos de Programación Orientada a Objetos en los estudiantes de la carrera de Computación de la Universidad Católica de Santiago de Guayaquil?

Evaluación del Problema

De los criterios definidos para la evaluación del problema, se escogerán los siguientes que se indican en la tabla:

Tabla 2 Evaluación del Problema

#	Criterios	Contexto
1	Delimitado	El problema está centrado específicamente en la Carrera de Computación de la UCSG, afectando a estudiantes y docentes de la asignatura de Programación Orientada a Objetos (POO) durante el periodo académico.
2	Claro	La definición del problema expone: la dificultad cognitiva en el aprendizaje de conceptos claves y la ineficiencia administrativa en la gestión de tutorías.
3	Relevante	La resolución de este problema es para la comunidad educativa, ya que la POO es un eje vital en la carrera; mejorar su aprendizaje y gestión impacta directamente en la reducción de estudiantes en reprobación.
4	Evidente	Como se indica en la ilustración # 1 las causas que se indican son claras, no requieren mayor detalle para poder definir el problema.
5	Concreto	En una línea se contextualiza el problema, de manera clara, concisa y precisa.
6	Factible	El proyecto es viable dado que se cuenta con los recursos tecnológicos (hardware/software), el acceso a la información institucional y el tiempo para desarrollar e implementar el prototipo propuesto.
7	Identifica productos esperados	Se define el resultado final como un prototipo de software (Tutor Inteligente) que ofrece soporte académico autónomo al estudiante.

El problema planteado en este trabajo es bajo el concepto de aquellas dificultades de aprendizaje de los estudiantes en la materia de POO y la ineficiente gestión de tutorías académicas que se mantiene en la carrera de Ingeniería en Ciencias de la computación de la UCSG.

Por tal motivo se seleccionaron los aspectos generales para evaluar esta problemática sus puntos más importantes y las cuales se detallan a continuación:

Delimitado: Al enfocarse en el período académico actual de la asignatura de POO en la Carrera de Ingeniería en Ciencias de la Computación de la

Universidad Católica de Santiago de Guayaquil (UCSG), involucrando específicamente a estudiantes y docentes de esta institución.

Claro: Es precisa y fácil de comprender, la cual se sustenta en una pregunta interrogativa que relaciona directamente la implementación de un tutor inteligente con la optimización de la gestión académica y el aprendizaje de POO

Evidente: Las manifestaciones del problema son observables y claras, evidenciadas en porcentajes de deficientes desempeños en POO debido a la alta complejidad de sus conceptos, así como en procesos administrativos manuales que limitan el seguimiento de tutorías.

Concreto: El problema se redacta de forma corta, precisa y directa, en una sola línea interrogativa que apunta hacia: la influencia de un prototipo en un contexto definido.

Relevante: Resolver este problema es crucial para la comunidad educativa de la UCSG, ya que POO es fundamental en la formación de ingenieros en computación, y su mejora reduce limitaciones en las tutorías.

Original: El enfoque es novedoso al integrar tutoría inteligente semántica ligada estrictamente al sílabo de POO

Contextual: El problema pertenece plenamente en el contexto educativo de la UCSG para la asignatura de POO, que resultan aprendizajes insuficientes ante la alta demanda estudiantil y la complejidad conceptual de temas.

Factible: El problema es factible dada la disponibilidad de recursos tecnológicos, acceso a datos institucional y tiempo para desarrollar el prototipo.

Variables: El problema identifica la variable independiente como el prototipo de tutor inteligente y la variable dependiente como el aprendizaje de conceptos al igual que la optimización de la gestión académica en estudiantes y docentes de la

carrera.

OBJETIVOS

Objetivo General

Desarrollar un prototipo de tutor inteligente utilizando Python, react y herramientas en la nube como Azure Open Ai e instancia de ProgreSQL con el fin de apoyar el proceso de enseñanza-aprendizaje y fortalecer la autonomía del estudiante por medio de la gestión autónoma y el registro de tutorías académicas en la asignatura de POO (programación orientada a objetos) de la Carrera de Computación en la Universidad Católica de Santiago de Guayaquil.

Objetivos Específicos

1. Analizar estructuradamente el silabo y los temas más importantes de la asignatura fundamentado estrictamente en el silabo oficial y sus contenidos claves.
2. Diseñar la estructura de datos y arquitectura web que permita la gestión Autónoma de Tutorías Académicas en las asignaturas de la Carrera de Computación.
3. Desarrollar el Prototipo de tutor inteligente para la Gestión Autónoma de Tutorías Académicas en las asignaturas de la Carrera de Computación.
4. Validar el prototipo de tutor Inteligente mediante pruebas funcionales y de usabilidad con estudiantes de la Facultad de Computación.

ALCANCES DEL PROBLEMA

Este trabajo de titulación se delimita al diseño y desarrollo para la implementación de un prototipo de tutor inteligente tipo chat, basado en una interfaz web para la gestión autónoma de tutorías en la asignatura de Programación Orientada a Objetos (POO) de la Carrera de Ingeniería en Ciencias de la Computación de la

UCSG.

Para ello se elaborará un prototipo funcional con búsqueda semántica que permite además la inspección de temas relacionados estrictamente con el sílabo de POO, con interacciones uno a uno entre el usuario y el tutor, sin sesiones compartidas, colaborativas ni multiusuario, se mantendrá el registro de consultas para evidencias del docente y métricas básicas para el uso académico.

No abarcará despliegue a producción, integración con LMS externos ni extensión a otras asignaturas o carreras, ya que estrictamente solo se afectará al módulo de POO y procesos administrativos de tutorías en la carrera de computación, sin impacto en otras facultades.

JUSTIFICACION E IMPORTANCIA

Este proyecto surge de la necesidad de abordar las limitaciones en el aprendizaje de POO, donde la alta comprensión de conceptos genera reprobaciones recurrentes en la materia, limitadas por tutorías presenciales y registros manuales ineficientes.

Su solución se radica en proporcionar una herramienta autónoma 24/7 que personaliza el soporte pedagógico y en la generación automatizadas a registros de interacciones, beneficiando directamente a estudiantes, docentes y administrativos de la carrera. Socialmente contribuye a formar ingenieros competentes en software, alineado con la actual innovación educativa.

- Implicaciones prácticas: Resuelve problemas reales como la accesibilidad limitada de tutorías, ofreciendo feedback inmediato y adaptativo vía IA, similar a ITS probados en algoritmos y bases de datos.
- Valor teórico: Enriquece el marco de sistemas de tutores inteligentes al integrar aplicaciones en educación superior en la carrera de computación

de la UCSG.

- Utilidad: El prototipo servirá como base escalable para UCSG, mejorando tasas de aprobación en POO, con impacto en calidad educativa.

PREGUNTA DE INVESTIGACIÓN

En la UCSG, las tutorías académicas de Programación Orientada a Objetos dependen de métodos manuales, limitados por horarios, alta demanda estudiantil y la complejidad de conceptos. Este contexto resalta la necesidad de herramientas autónomas que complementen el apoyo docente sin reemplazarlo, basándose en evidencias de sistemas de tutores inteligentes que optimizan el aprendizaje mediante interacciones personalizadas.

Por lo tanto, se deriva la siguiente Pregunta de Investigación: ¿De qué manera el desarrollo de un prototipo de tutor inteligente podrá influir en la optimización de la gestión académica y en el fortalecimiento del aprendizaje de los conceptos de Programación Orientada a Objetos en los estudiantes de la carrera de Computación de la Universidad Católica de Santiago de Guayaquil?

Esta interrogante interpreta el análisis causa-efecto documentado en la UCSG y se alinea con predicciones estudiantiles y en evidencias de docentes mediante búsquedas semánticas ligadas al sílabo.

VARIABLES DE LA INVESTIGACIÓN

Para este estudio se definieron las variables de forma específica para medir el impacto del prototipo de tutor inteligente en el contexto de POO en la UCSG, derivando directamente de la pregunta de investigación y objetivos del prototipo. Se identifican dos principales: la variable independiente: prototipo de tutor inteligente, y las variables dependientes: aprendizaje de conceptos de POO y optimización de gestión académica, con mediciones vía encuestas.

Variables Independientes: Python, (Backend), React (Frontend), Azure OpenAi (motor de IA), PostgreSQL (Estancia).

Variable Dependiente 1: Prototipo de Tutor Inteligente.

Capítulo II

Marco Teórico

Antecedentes del estudio

Fundamentación teórica

Este marco teórico aborda toda la información recolectada de la investigación teórica en cuanto a procesos y conceptos generales, que se requieren para la puesta en marcha del tutor inteligente, así como también conceptos relacionados con la asignatura que este tutor inteligente abordara en su funcionamiento.

En este contexto, el marco teórico cumple con la función de orientar el estudio, revisar los conceptos claves y definir los enfoques existentes relacionados a la problemática planteada y abordada.

En este escenario, las sesiones de tutoría académica se establecen como una herramienta esencial para abordar las variaciones en el ritmo de aprendizaje y potenciar la asimilación de los temas.

Las tecnologías inteligentes y su incorporación en el ámbito educativo han permitido replantear modelos tradicionales en el tema de las tutorías, estos a su vez han dado origen a los que se denominan tutores inteligentes.

Etimología y Concepto.

Los desafíos de la enseñanza de Ciencias de la Computación para estudiantes y profesores se pueden percibir al analizar las diversas disciplinas en el área. Las dificultades que los estudiantes experimentan durante el proceso de enseñanza-aprendizaje en las disciplinas de algoritmos y estructuras de datos son una preocupación importante para los educadores de todo el mundo. (Rodrigo Elias Francisco, 2022)

El Sistema de Tutoría Inteligente (STI) es un software educativo destinado para

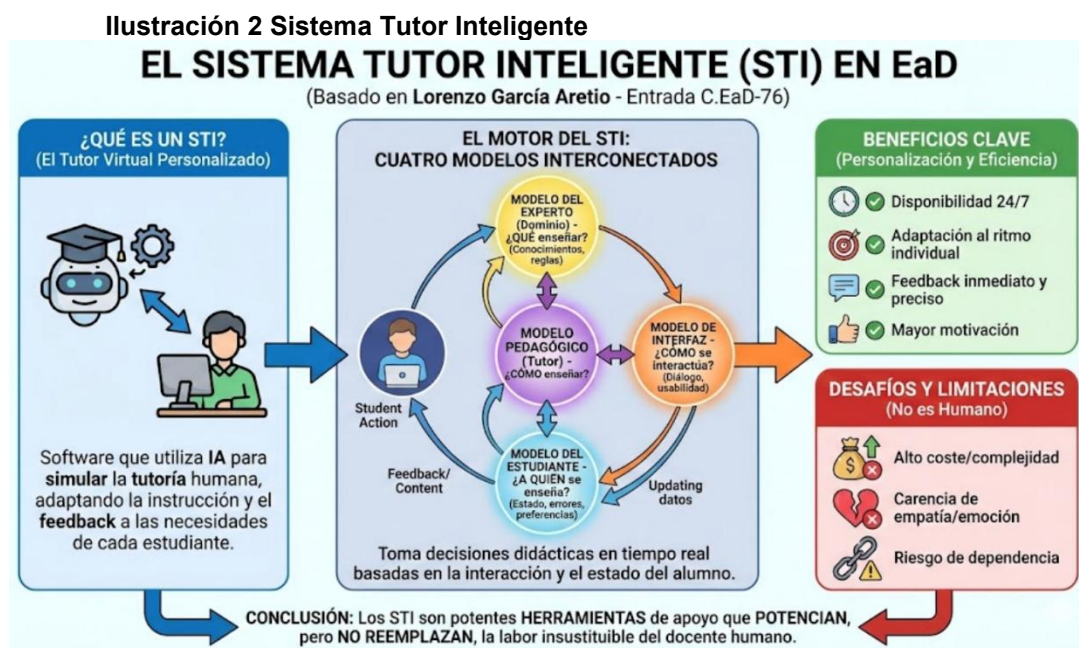
adaptar la enseñanza al perfil del estudiante. Utilizando Inteligencia Artificial (IA), estos sistemas pueden contribuir a la automatización de etapas del proceso de enseñanza-aprendizaje en esta área. Por ejemplo, apoyar a un estudiante en la solución de problemas que involucran práctica y razonamiento abstracto puede ser posible gracias a estos sistemas (Ali Alkhatlan, 2018)

Un estudiante de nuevo ingreso que toma un primer curso en algoritmos y programación computacional desconoce varias temáticas relacionadas con entender un texto, analizar y resolver un problema matemático o plantear una estrategia de solución para llevar a cabo la implementación de un algoritmo a una problemática presentada. El sector educativo está cambiando con la evolución e implantación de nuevas tecnologías. El éxito de superar estos obstáculos está relacionado con dos elementos principales en el proceso de enseñanza-aprendizaje: el estudiante y el profesor. El primero al tener tenacidad e iniciativa de aprender, entendiendo el valor que tiene el aprendizaje para el éxito de emplearse como experto en la industria del software y el segundo al identificar las fortalezas y debilidades del alumno para plantearle un plan de acción que permita la autorrealización del alumno, y que aumente la tolerancia a la frustración de este. (Chávez, 2025)

Frente a este escenario, un Sistema Tutor Inteligente enfocado en programación orientada a objetos podría ser una solución efectiva para mejorar el aprendizaje. Un STI diseñado para programación orientada a objetos puede asistir a los estudiantes en la resolución de consultas, brindar pistas ante errores y reforzar la comprensión con retroalimentación adaptativa (Sophia Yang, 2021)

A partir de estos avances, los tutores inteligentes han evolucionado hacia sistemas aún más complejos que integran modelos para estudiantes, el manejo adecuado

de la asignatura con su respectivo aprendizaje y de la estrategia pedagógica.



Nota: El Sistema Tutor Inteligente (STI) constituye una de las materializaciones más sólidas y con mayor recorrido de la inteligencia artificial en el ámbito educativo.

El STI nace con una vocación estrictamente pedagógica. Su misión intrínseca es:

- Diagnosticar el estado del aprendiz.
- Decidir cuál es la siguiente mejor acción educativa para él
- Sustener un diálogo formativo ajustado a metas de aprendizaje y registrando las evidencias de su progreso. (Aretio, 2025)

Funcionalmente hablando, los tutores inteligentes pueden clasificarse en tipos distintos según su enfoque e implementación. Entre los que más destacan para la enseñanza de la programación se encuentran aquellos basados en reglas, que guían al estudiante en secuencias predefinidas de aprendizaje; los tutores adaptables, que ajustan el contenido y la retroalimentación en función del desempeño del estudiante; y aquellos orientados a la resolución de problemas, los cuales priorizan el aprendizaje activo mediante la práctica y la examinación de errores. (Nkambou, 2010).

Herramientas de Desarrollo

El prototipo se desarrolló bajo una arquitectura web cliente-servidor, donde se logró integrar tecnologías modernas de desarrollo frontend, backend, bases de datos y servicios en la nube.

Lenguajes y tecnologías de desarrollo

Para el desarrollo de la interfaz de usuario se emplearon tecnologías que permiten la visualización e interacción con el sistema. Facilitando el acceso y una mejor adaptabilidad a diferentes dispositivos, las cuales se detallan a continuación:

HTML

Por sus siglas HyperText Markup Language, es el lenguaje estándar que se utiliza en la web para poder definir una estructura en una página web y el poder visualizar su contenido en forma de texto, como de igual forma poder implementar objetos como imágenes. (Herrador)

Este lenguaje nos permitió organizar los elementos visuales de forma más organizada para poder gestionarla en cada sección del sistema y los contenidos del prototipo.

CSS

Por sus siglas Cascading Style Sheets, es un lenguaje que se utiliza para la presentación de una página web, es decir que permite separar el contenido y estructura de la página web para una mejor estética, flexibilidad y una mejor reutilización de código. (Casado, 2023)

Se implementó para su diseño y flexibilidad al momento de gestionar cada sección del sistema y que todo sea más ordenado para la interacción con el usuario.

Tecnologías de desarrollo backend

Para el desarrollo del proyecto se gestionó la selección de un stack tecnológico que

sea escalable y con alto rendimiento para las operaciones de concurrencia que se llevará a cabo en el sistema.

A continuación, se describen las herramientas seleccionadas para la implementación en esta capa de backend:

Python

Es un lenguaje de programación caracterizado por ser versátil por su variedad de campos para el desarrollo de páginas web, además que es eficiente y tiene una facilidad de aprendizaje para su implementación. (Sucari, 2024)

Este lenguaje tiene su capacidad de poder ejecutarse ante múltiples plataformas debido a su sintaxis clara y legible para una estructura con fácil comprensión. (Sucari, 2024)

FastAPI

Es un framework web moderna y de alto rendimiento para la construcción de APIs con Python, cuya función es la centralizar datos de los errores de escritura y generar de forma automática la documentación de los métodos implementados. (FastAPI, s.f.)

Se caracteriza por ser fácil de desarrollo, rapidez en su ejecución y es flexible al momento de determinar parámetros para así poder reducir posibles errores o duplicación de código. (FastAPI, s.f.)

Base de datos

Para la capa de almacenamiento de la información se empleó una base de datos relacional, ya que al utilizar este tipo de base de datos garantiza la integridad y la organización de los datos en el sistema. Para ello se consideró el siguiente sistema:

PostgreSQL

Es un sistema de base de datos relacional de objetos de código abierto con

numerosas funciones que permiten almacenar y escalar de forma segura las cargas de trabajo de datos complejas. (PostgreSQL, 2025)

Este sistema está diseñada para ayudar a los desarrolladores a gestionar de mejor forma aplicaciones y también administrar la integridad de los datos ante entornos tolerantes a fallos. (PostgreSQL, 2025)

Para la elección de esta base de datos relacional se consideró los criterios de escalabilidad, compatibilidad, rendimiento y seguridad de integración con las tecnologías de desarrollo web e inteligencia artificial. Se presenta una tabla comparativa para su definición:

Tabla 3 Criterios Comparativos de Base de Datos Relacionales

Criterio	PostgreSQL	MySQL	SQLite
Escalabilidad	Permite manejar múltiples usuarios y grandes volúmenes de datos sin afectar el rendimiento del sistema.	Soporta crecimiento de datos, pero puede requerir ajustes adicionales.	Está diseñado para aplicaciones pequeñas y no para sistemas con usuarios simultáneos.
Integración con Python	Posee librerías y conectores que permiten una integración eficiente con frameworks como FastAPI.	Cuenta con conectores funcionales para Python, aunque con menor flexibilidad	Se integra fácilmente con Python, pero está orientado a proyectos limitados.
Manejo de grandes volúmenes de datos	Gestiona grandes cantidades de información y consultas complejas con alto rendimiento.	Puede manejar volúmenes considerables de datos, aunque con menor optimización.	Presenta restricciones cuando se requiere almacenar grandes cantidades de datos o múltiples accesos.
Seguridad de datos	Ofrece control avanzado de accesos, integridad y mecanismos de seguridad robustos.	Cuenta con herramientas de seguridad básicas.	No está diseñado para entornos con múltiples usuarios ni altos niveles de seguridad.
Soporte en la nube (Azure)	Puede implementarse fácilmente en servicios de nube como Microsoft Azure con alta disponibilidad.	También es soportado por plataformas cloud, aunque con menor optimización en algunos servicios.	No está orientado a implementaciones en la nube ni a sistemas distribuidos.

Infraestructura en la nube

Este sistema se logró implementar en un entorno de computación en la nube, con el objetivo de poder garantizar el acceso remoto y la disponibilidad continua del sistema. El uso de esta plataforma en la nube permitió alojar la aplicación web, la base de datos y los servicios de inteligencia artificial para la gestión de pruebas e interacción con el usuario.

Microsoft Azure

Permite crear, lanzar y gestionar con facilidad aplicaciones en una red global de centros de datos en el sistema de Microsoft, al ser una plataforma en la nube de confianza ofrece una simplificación en la administración de la infraestructura local. (Microsoft Azure., s.f.)

Para la elección de esta plataforma se consideró varios factores que permitan una gestión eficiente del sistema, las cuales se detallan en la siguiente tabla comparativa:

Tabla 4 Comparación de plataformas de computación en la nube

Criterio	Microsoft Azure	AWS	Google Cloud
Facilidad de implementación	Permite desplegar aplicaciones web y bases de datos de forma rápida.	Ofrece gran variedad de servicios, pero su configuración inicial puede ser compleja.	Requiere configuración técnica para integrar los servicios del sistema.
Tiempo de despliegue para prototipos	Permite implementar aplicaciones en menor tiempo.	Requiere configuración más detallada antes del despliegue.	El proceso de implementación puede requerir mayor ajuste técnico.
Compatibilidad académica	Es utilizado en entornos educativos y ofrece servicios accesibles para proyectos académicos.	Es robusto, pero su complejidad puede ser mayor para proyectos académicos.	Requiere mayor configuración técnica para prototipos educativos.

Integración de inteligencia artificial

Este sistema de Tutor Inteligente implementa modelos de inteligencia artificial que se encuentra orientado precisamente a la interacción que va a tener el estudiante con el sistema y de la misma forma en el apoyo en la resolución de las interrogantes

que existan en relación con la asignatura de Programación Orienta a Objetos.

Estos componentes permiten:

- Procesar las preguntas en lenguaje natural.
- Generar respuestas.
- Proporcionar la retroalimentación.
- Simular un proceso de tutoría digital.

En la práctica, la IA en la educación se manifiesta en las herramientas diseñadas para personalizar el aprendizaje en el estudiante, adaptándose a las necesidades individuales que conlleva. En resumen, la IA representa una metodología tecnológica destinada a la transformación de oportunidades ante los desafíos para la comunidad educativa. (Ciencia Latina Revista Científica Multidisciplinar)

Control de versiones

Para el sistema del tutor inteligente se debe considerar la gestión del código fuente que permita realizar el control de cambios, versionamiento del sistema, registro de modificaciones.

Esta herramienta se empleó con el fin de registrar, organizar y controlar los cambios que se realizaron durante el desarrollo de este, se tomó en cuenta el siguiente:

Git

Es un sistema de control de versiones distribuido, es decir se convierte en un clon local del proyecto en un repositorio para facilitar el trabajo sin conexión o de forma remota. Su flexibilidad y popularidad convierte esta herramienta a una mejor opción para cualquier equipo. (Azure DevOps. Microsoft Learn, s.f.)

Capítulo III

Metodología De La Investigación

La presente investigación se desarrolló con un enfoque cuantitativo, ya se que emplearon procesos sistemáticos y empíricos para generar una respuesta al objeto de estudio que se planteó para la necesidad y aceptación de un tutor inteligente para la asignatura de Programación Orientada a Objetos. (Pino, 2022)

El enfoque cuantitativo permitió medir las percepciones, niveles de dificultad y el uso de las herramientas tecnológicas en el proceso de aprendizaje de los estudiantes, mediante datos numéricos que posteriormente fueron analizados e interpretados.

Según (Hernández-Sampieri, 2022) la investigación cuantitativa se caracteriza por la recolección de datos para con ello poder probar supuestos o responder preguntas de investigación mediante el análisis estadístico.

El trabajo investigativo busca responder a la siguiente pregunta:

¿De qué manera el desarrollo de un prototipo de tutor inteligente podrá influir en la optimización de la gestión académica y en el fortalecimiento del aprendizaje de los conceptos de Programación Orientada a Objetos en los estudiantes de la carrera de Computación de la Universidad Católica de Santiago de Guayaquil?

El estudio se presenta como un proyecto factible, justificado a que se propone el desarrollo de un prototipo tecnológico que busca solucionar una problemática actual académica identificada en el proceso de aprendizaje de la

asignatura de Programación Orientada a Objetos. En este caso, el estudio planteó el diseño de un tutor inteligente basado en inteligencia artificial como una herramienta de apoyo académico.

Tipo de Investigación

El estudio se desarrolló dentro de dos categorías de investigación complementarias:

Investigación Descriptiva

Este tipo de investigación cumple con poder determinar la situación actual del proceso académico en la asignatura de Programación Orientada a Objetos, se identifican y detallan las dificultades que se enfrenta bajo los conceptos complejos para los estudiantes y las limitaciones manuales de las tutorías por los docentes.

Investigación Aplicada

El objetivo de este proyecto no es solo generar nuevos conocimientos teóricos, sino el de poder desarrollar una solución tecnológica tangible que pueda resolver esta problemática de forma evolutiva.

Enfoque de la Investigación

Como se mencionó anteriormente este estudio se adopta a un enfoque cuantitativo, porque se utilizó la recolección de datos para probar hipótesis con el análisis estadístico, para establecer patrones de comportamiento y probar teorías.

En el marco de este proyecto, se cuantificarán variables como el nivel de satisfacción con las tutorías actuales, la frecuencia de dudas en temas del sílabo, la aceptación y la receptividad tecnológica del prototipo propuesto.

Población y Muestra

Población

Para la población de estudio se consideró aquellos estudiantes matriculados en la carrera de Computación de la UCSG que se encuentran cursando o han cursado recientemente la asignatura de Programación Orientada a Objetos. Esta población fue seleccionada debido a que los estudiantes poseen previos conocimientos en programación y se encuentran directamente relacionados con el objeto de estudio.

Estos estudiantes presentan las siguientes características:

- Formación en programación
- Necesidad de reforzar contenidos de POO
- Uso de herramientas tecnológicas para aprendizaje

Tabla 5 Población de estudio

<i>Población</i>	<i>N</i>
<i>Estudiantes de Computación</i>	50
<i>Total</i>	50

Muestra

Debido a que la población encuestada es reducida, se trabajó con la totalidad de los estudiantes disponibles, por lo que se utilizó un muestreo censal. El tipo de muestreo que se detalla a continuación permite obtener información directa de todos los elementos que conforman la población, evitando errores de estimación.

La muestra quedó conformada por:

Tabla 6 Muestra del estudio

<i>Muestra</i>	<i>N</i>
<i>Estudiantes encuestados</i>	50
<i>Total</i>	50

Técnicas e Instrumentos de Recolección de Datos

Para poder analizar la información y poder justificarlas, se utilizaron las siguientes técnicas:

Técnica:

La encuesta ha servido para obtener información estandarizada de un número de sujetos de estudio de forma eficiente.

Instrumento:

El cuestionario, para ello se diseñó un cuestionario estructurado con preguntas cerradas utilizando una Escala de Likert (1 al 5).

Este instrumento evalúa: dificultades de aprendizaje en POO, eficiencia de la gestión actual de tutorías y la intención de uso de herramientas autónomas.

ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

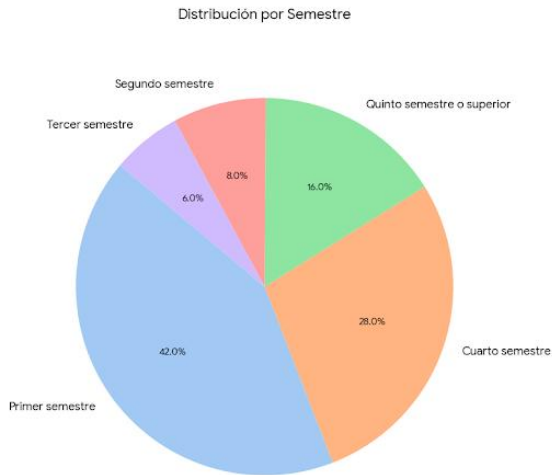
La encuesta se aplicó a una muestra representativa de estudiantes del programa de Ciencias de la Computación de la Universidad Católica de Santiago de Guayaquil. El objetivo del instrumento fue diagnosticar dificultades de aprendizaje en el curso de Programación Orientada a Objetos (POO) y validar la aceptación de una herramienta tecnológica de apoyo como es el Tutor Inteligente.

Los datos fueron procesados mediante estadística descriptiva, y se presentan a continuación en tablas de distribución de frecuencias y gráficos de porcentajes para su respectivo análisis:

Como se muestra en la siguiente ilustración, la mayoría de la población encuestada se encuentra en las primeras etapas de sus estudios, siendo el primer semestre un 42% y el cuarto semestre 28% los grupos predominantes.

Esto indica que la muestra refleja tanto las perspectivas de los estudiantes principiantes como de aquellos que ya poseen un conocimiento más profundo de las asignaturas de programación.

Ilustración 3 Distribución de estudiantes por semestre

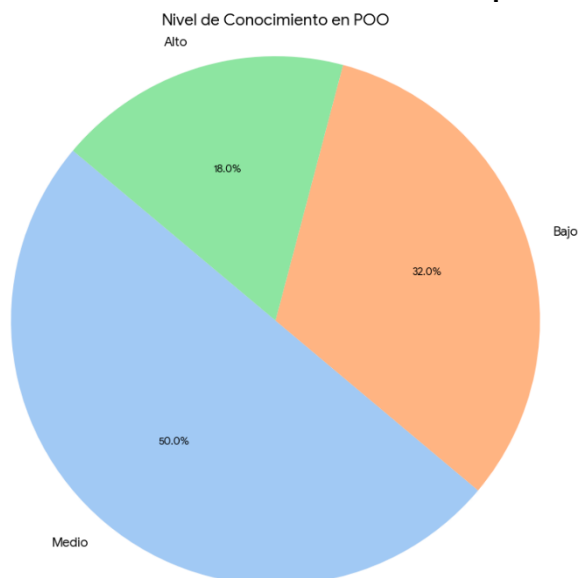


Nota: Gráfico estadístico de la distribución de estudiantes por semestres.

En cuanto al nivel de dominio de la materia como se detalla en el siguiente gráfico, se consideró que el 50% de los encuestados se encuentran en un nivel intermedio, mientras que el 32% lo considera bajo. Solo un porcentaje del 18% se percibe con un nivel alto.

Estos datos revelan una importante brecha de conocimiento que justifica la implementación de herramientas de refuerzo académico.

Ilustración 4 Nivel de conocimiento auto percibido en POO

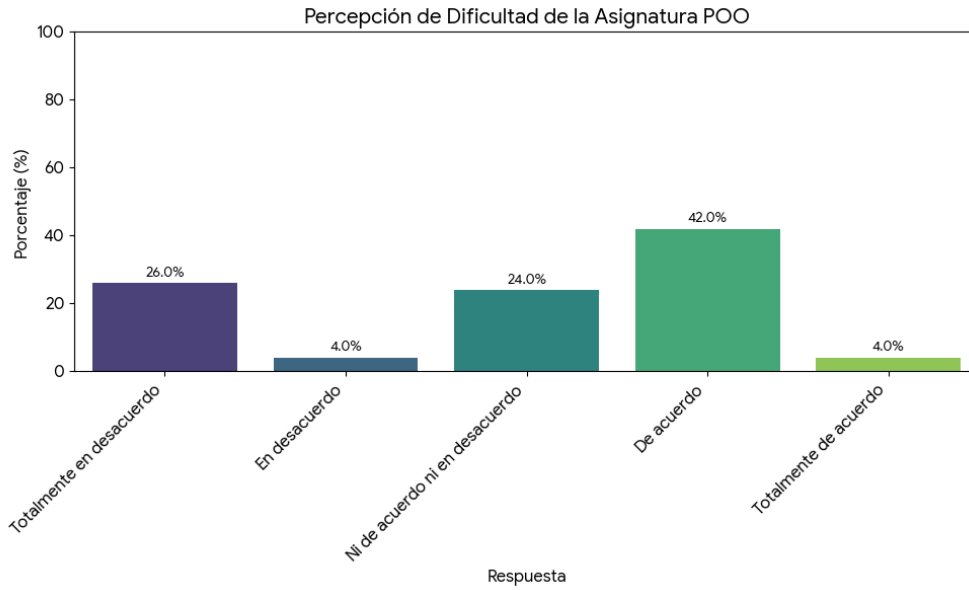


Nota: Gráfico estadístico de la distribución del nivel de conocimiento de los estudiantes en POO

En cuanto a la complejidad del tema detallado en el siguiente gráfico, se tomó en cuenta que el 46% de los estudiantes, combinando las respuestas "De acuerdo" y "Totalmente de acuerdo", afirma que la programación orientada a objetos presenta un alto nivel de dificultad.

Sin embargo, un significativo 26% de los estudiantes que están "Totalmente en desacuerdo" sugiere que la dificultad puede ser subjetiva y depender de los conocimientos previos del estudiante, lo que refuerza la necesidad de una herramienta que ofrezca un aprendizaje personalizado.

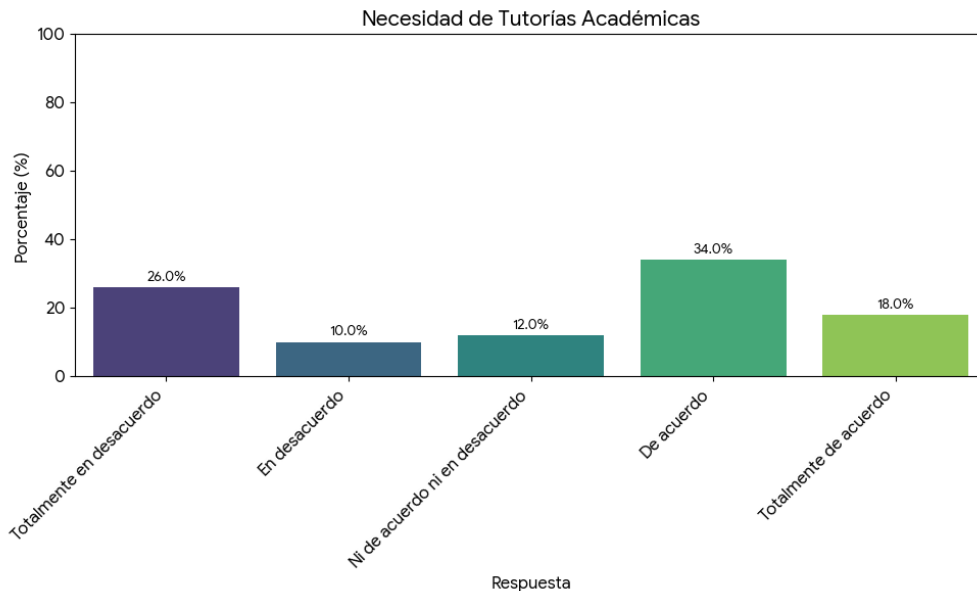
Ilustración 5 Percepción de dificultad de la asignatura



Nota: Gráfico estadístico de los resultados sobre la dificultad de la asignatura para los estudiantes

La siguiente ilustración revela que el 52% de los estudiantes han requerido tutoría académica para comprender aún más la asignatura de POO. Esto valida de que el apoyo extracurricular es una necesidad latente para más de la mitad de la población estudiantil.

Ilustración 6 Necesidad de tutorías académicas

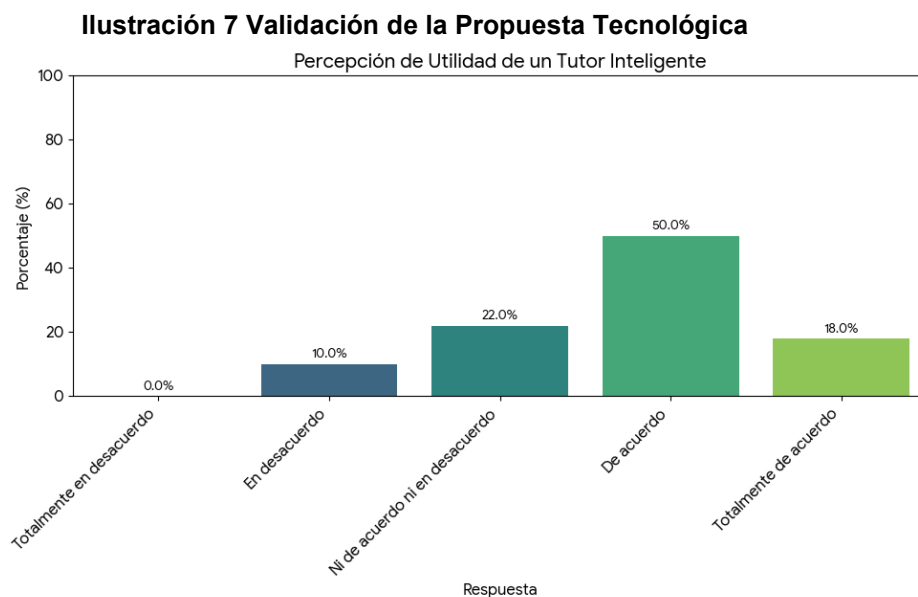


Nota: Barras estadísticas de los resultados a la necesidad de tutorías.

Al preguntar sobre la solución tecnológica propuesta, los resultados son contundentes ya que el 68% de los encuestados:

- 50% de acuerdo
- 18% totalmente de acuerdo

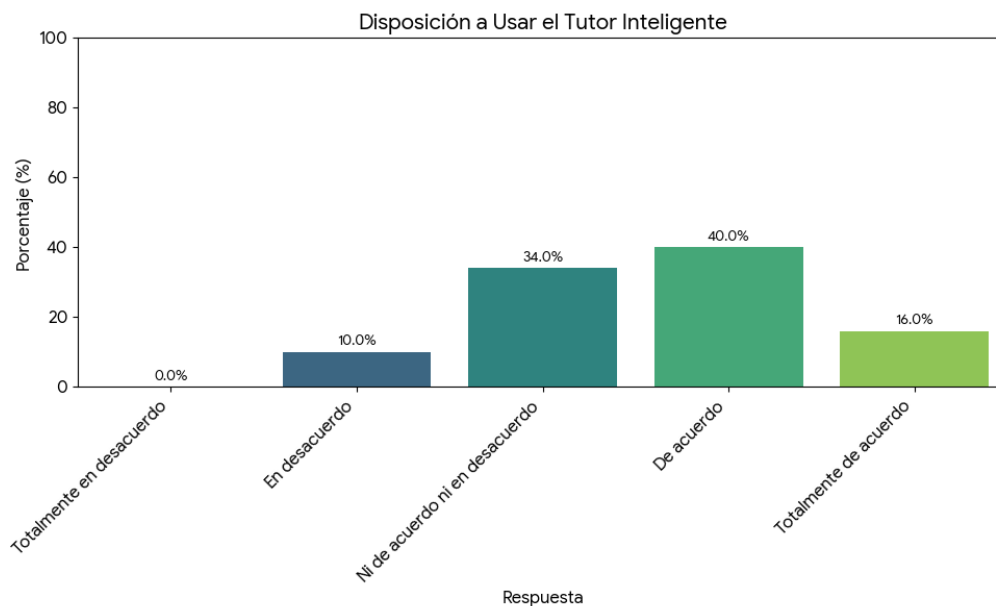
Cree que un Tutor Inteligente sería una herramienta útil. Solo el 10% discrepó. Este alto porcentaje de aceptación demuestra la viabilidad del proyecto desde la perspectiva del estudiante. El siguiente gráfico detalla los resultados:



Nota: Gráfico estadístico de los resultados sobre la utilidad que tendría el tutor inteligente ante los estudiantes.

Finalmente, la siguiente ilustración muestra que el 56% de los estudiantes estaría dispuesto a usar activamente el Tutor Inteligente como apoyo en la asignatura. Cabe destacar que el 34% se mantiene neutral: "Ni de acuerdo ni en desacuerdo", lo que representa que este grupo podría convertirse en usuarios activos si la herramienta resulta eficaz en las interacciones iniciales.

Ilustración 8 Disposición de uso de la herramienta



Nota: Gráfico de los resultados estadísticos sobre la disposición en usar el tutor inteligente.

Conclusiones del análisis

El análisis estadístico realizado a partir de la muestra de 50 estudiantes nos permite obtener conclusiones relevantes en relación con el proceso de enseñanza-aprendizaje además de confirmar que existe una necesidad de incluir e incorporar herramientas tecnológicas de apoyo entonces concluimos que:

- **Existe una necesidad real:** Este resultado nos permite evidenciar que más de la mitad del alumnado requiere apoyo adicional y percibe la asignatura como considerablemente difícil.
- **El nivel es mejorable:** Con un 82% de estudiantes en niveles "Bajo" o "Medio", existe un amplio margen para que el Tutor Inteligente tenga un impacto positivo en el rendimiento académico.
- **Alta aceptación tecnológica:** La propuesta del Tutor Inteligente cuenta con el apoyo de casi el 70% del alumnado, lo que valida la relevancia de desarrollar este prototipo como Proyecto de Integración Curricular.

Metodología de Desarrollo de Software

Para la construcción del prototipo de Tutor Inteligente, se ha seleccionado la metodología en Cascada, este modelo de ciclo de vida de software se alinea con

los requerimientos académicos del presente proyecto, ya que mantuvo una documentación ajustada y una planificación secuencial.

El desarrollo se estructura en las siguientes fases:

- **Análisis de Requisitos:** Recopilación y definición de las necesidades pedagógicas.
- **Diseño del Sistema:** Definición de la arquitectura, diseño de la base de datos y diagramas de flujo de la interacción con la IA.
- **Implementación:** Desarrollo del backend en Python con FastAPI y la integración de los servicios de Azure y OpenAI.
- **Pruebas:** Ejecución de pruebas de integración para asegurar que el tutor responde correctamente según el sílabo y que el registro de tutorías se almacena sin errores.
- **Mantenimiento:** Propuesta de plan para futuras actualizaciones del sílabo o mejoras en el modelo.

Capítulo IV

Propuesta Tecnológica

HERRAMIENTAS DE DESARROLLO

En este capítulo se presenta la propuesta tecnológica la cual está diseñada para la creación de un prototipo de tutor inteligente que se enfoca en todo su proceso a la gestión autónoma de tutorías en la materia de POO (programación orientada a objetos).

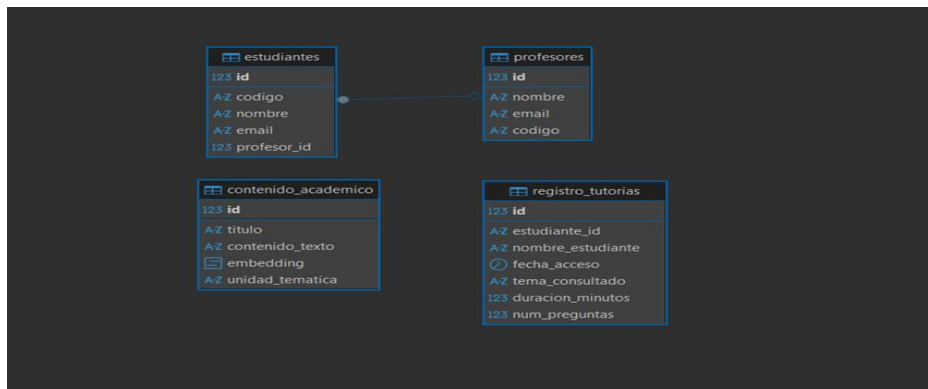
En esta parte se describen los elementos técnicos, arquitecturales, y funcionales que respaldan esta solución, así mismo los criterios que se tienen en cuenta para su diseño y ejecución que van alineados a los objetivos fijados en la investigación. Para poder desarrollar este capítulo se consideró las decisiones técnicas para la arquitectura del sistema, en ello se tomó en cuenta los elementos como el entorno de desarrollo, lenguajes de programación, frameworks y recursos tecnológicos requeridos para el buen funcionamiento del tutor inteligente.

En consecuencia, se busca impartir una perspectiva clara y ordenada de la propuesta, facilitando la comprensión de cómo las herramientas de desarrollo ayudan en la implementación de una solución tecnológica destinada a mejorar la gestión, supervisión y registro oportuno de las tutorías académicas para la asignatura de programación orientada a objetos.

Estructura de la base de datos.

La siguiente ilustración presenta el modelo lógico de la base de datos lo que a su vez sustenta el funcionamiento del tutor y este se compone por 4 entidades que son:

Ilustración 9 Estructura de la base de datos.



Estudiantes: Este almacena la información básica de los alumnos, incluyendo un identificador único, código institucional, nombre y correo electrónico. Además, incorpora una clave foránea que permite asociar a cada estudiante con un profesor tutor, estableciendo una relación directa entre ambos actores del proceso académico. Crea un vínculo para el seguimiento personalizado

Profesores: Contiene los datos identificativos del docente responsable de las tutorías, posee nombre, correo electrónico y código institucional. Permite gestionar a los tutores académicos y vincularlos con los estudiantes asignados, ayuda a la organización y administración de las tutorías que van quedando registradas.

Contenido académico: Aquí van los recursos didácticos utilizados por el tutor inteligente. En ella se almacena información relacionada con los temas abordados en la asignatura, incluyendo el título del contenido, el texto académico asociado y la unidad temática correspondiente. Adicionalmente, se incorpora un campo de tipo *embedding*, ósea se encarga de la representación semántica para apoyar los procesos de búsqueda, recomendación o respuesta inteligente del sistema, reforzando el carácter adaptativo del tutor.

Registro de tutorías: Permite almacenar información detallada de cada interacción entre el estudiante y el tutor inteligente.

Script de ingesta de datos

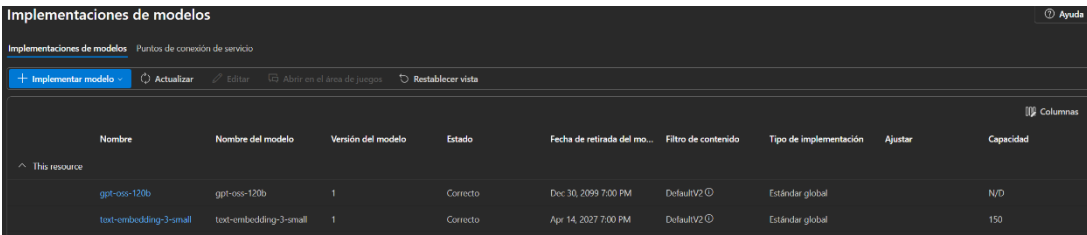
Ilustración 10 Script de ingesta de datos.

```
1 def procesar_e_ingresar_pdfs(folder_path):
2     try:
3         # Conectar a la base de datos
4         conn = psycopg2.connect(**DB_CONFIG)
5         cur = conn.cursor()
6         print("✅ Conexión exitosa a Azure PostgreSQL")
7
8         # Configurar el fragmentador de texto
9         # Usamos Recursive para mantener párrafos con sentido
10        text_splitter = RecursiveCharacterTextSplitter(
11            chunk_size=800, chunk_overlap=80)
12
13        for filename in os.listdir(folder_path):
14            if filename.endswith(".pdf"):
15                print(f"📄 Procesando: {filename}...")
16                path = os.path.join(folder_path, filename)
17                reader = PdfReader(path)
18
19                full_text = ""
20                for page in reader.pages:
21                    full_text += page.extract_text()
22
23                chunks = text_splitter.split_text(full_text)
24
25                for chunk in chunks:
26                    # Generar vector
27                    embedding = get_embedding(chunk)
28
29                    # Insertar en la tabla que creamos (Objetivo B y D)
30                    query = """
31                    INSERT INTO contenido_academico (titulo, contenido_texto, embedding, unidad_tematica)
32                    VALUES (%s, %s, %s, %s)
33                    """
34                    # Intentamos inferir la unidad por el nombre del archivo
35                    cur.execute(query, (filename, chunk,
36                                     embedding, filename.split('.')[0]))
37
38                    conn.commit()
39                    print(f"✅ {filename} cargado correctamente.")
40
41        cur.close()
42        conn.close()
43        print("🎉 Ingesta finalizada con éxito.")
```

La Ilustración correspondiente al módulo de ingesta y procesamiento de contenido académico del prototipo de tutor inteligente. El componente automatiza la carga de documentos en formato PDF, este procesa su contenido textual lo almacena en la base de datos junto con representaciones semánticas para que el tutor inteligente las use.

Modelos usados con el programa de estudiantes de Azure.

Ilustración 11 Implementación de modelo.



Nombre	Nombre del modelo	Versión del modelo	Estado	Fecha de retirada del mo...	Filtro de contenido	Tipo de implementación	Ajustar	Capacidad
gpt-oss-120b	gpt-oss-120b	1	Correcto	Dec 30, 2099 7:00 PM	DefaultV2	Estándar global		N/D
text-embedding-3-small	text-embedding-3-small	1	Correcto	Apr 14, 2027 7:00 PM	DefaultV2	Estándar global		150

Esta Ilustración pertenece a los modelos principales, implementados y disponibles para su uso. El primero corresponde a un modelo de lenguaje de propósito general, orientado a la generación de texto y a la interacción conversacional con los estudiantes. Este modelo constituye el núcleo del tutor inteligente, ya que permite interpretar consultas relacionadas con la asignatura de Programación Orientada a Objetos y generar respuestas contextualizadas a partir del conocimiento previamente cargado en el sistema.

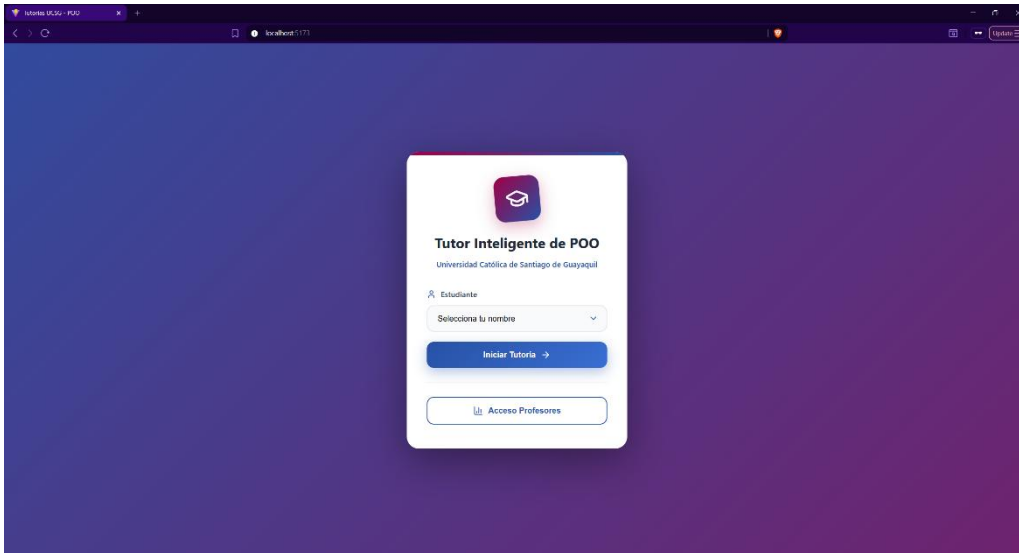
El segundo modelo implementado está orientado a la generación de embeddings o representaciones vectoriales del texto. Este tipo de modelo cumple una función esencial en el procesamiento del contenido académico, ya que permite transformar documentos y consultas en vectores semánticos que facilitan la búsqueda, comparación y recuperación de información relevante.

En conjunto, esta infraestructura de modelos de inteligencia artificial constituye un componente clave de la propuesta tecnológica, ya que habilita las capacidades cognitivas del tutor inteligente y asegura un procesamiento eficiente y confiable de la información académica. Su integración con la base de datos y los módulos de gestión de tutorías permite ofrecer un servicio de acompañamiento autónomo, adaptativo y alineado con las necesidades de los estudiantes de la asignatura de Programación Orientada a Objetos.

Pantalla de inicio

Esta ilustración es perteneciente a la pantalla de inicio del tutor inteligente los estudiantes ya están registrados por orden de nombre correspondiente al semestre en curso.

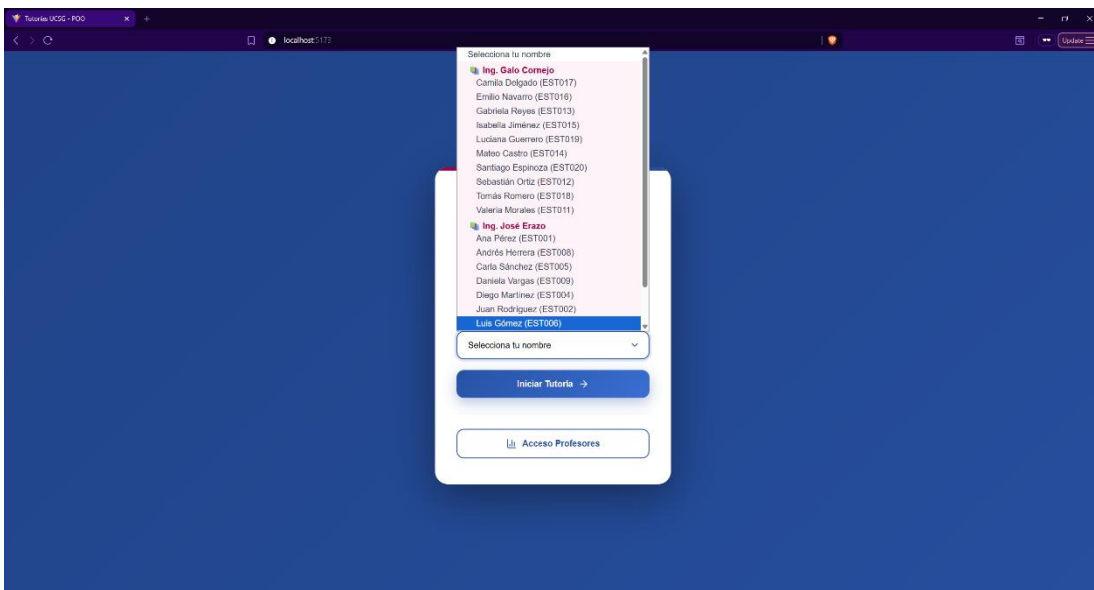
Ilustración 12 Pantalla de inicio



Inicio: selección de participantes:

En la ilustración se muestra una de las funcionalidades de el tutor la cual es clasificar por clases (a fecha de este trabajo de integración curricular 2 paralelos) con los respectivos profesores y alumnos asignados en la materia de programación orientada a objetos a la fecha de la realización de estas capturas de evidencia:

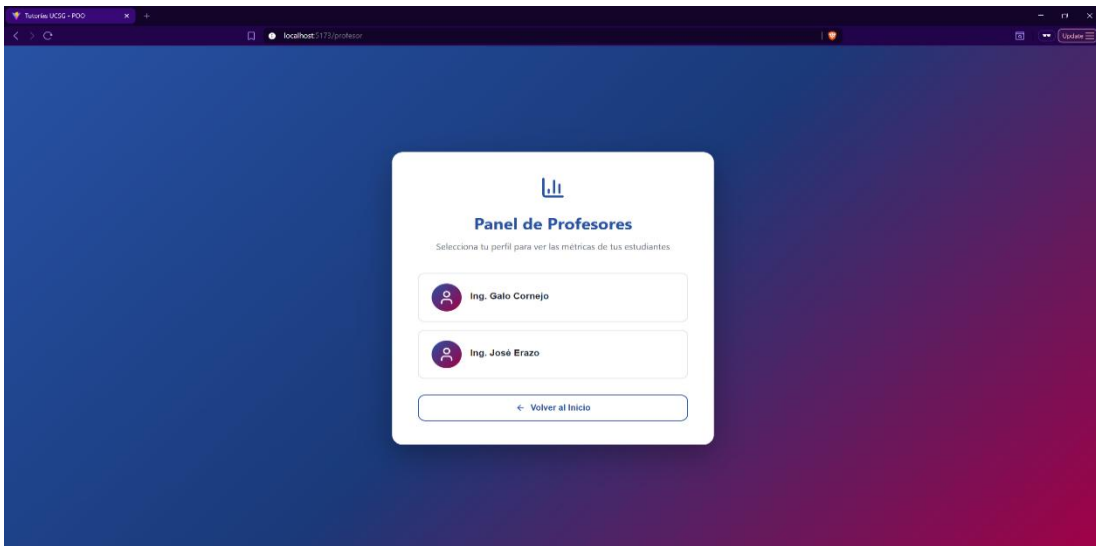
Ilustración 13 Funcionalidades



Dashboard de profesores

El tutor inteligente posee un dashboard para los profesores correspondientes asignados a cada paralelo (a fecha de este trabajo de integración curricular 2):

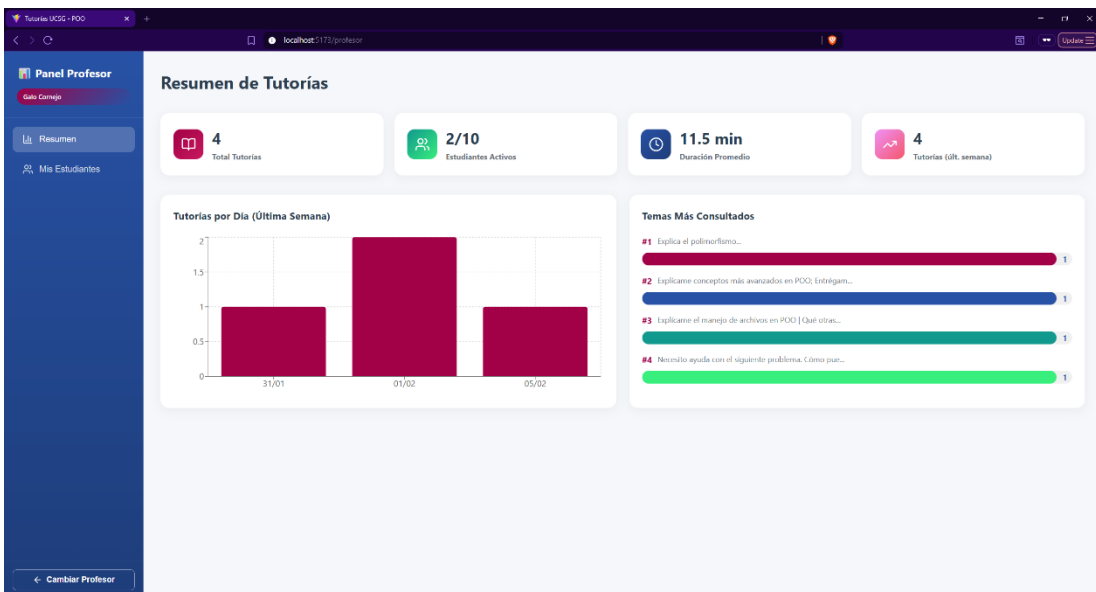
Ilustración 14 Dashboard de profesores



Descripción del panel de gestión y visualización de tutorías

La imagen muestra la interfaz del panel de control del tutor académico, correspondiente al prototipo de tutor inteligente desarrollado para la gestión autónoma de tutorías en la asignatura de Programación Orientada a Objetos.

Ilustración 15 Descripción del panel de gestión y visualización de tutorías



Este panel está diseñado para proporcionar al docente una visión clara, sintetizada y en tiempo real de la actividad tutorial registrada por el sistema.

Conclusiones

Tras completar el desarrollo del “Prototipo de Tutor Inteligente” y analizar los datos obtenidos en la investigación de campo del programa de Ciencias de la Computación de la Universidad Católica de Santiago de Guayaquil (UCSG), se presentan las siguientes conclusiones:

Se logró un entendimiento profundo de la asignatura al desglosar minuciosamente el sílabo oficial y sus temas centrales. Este proceso de análisis permitió que el proyecto no se base en suposiciones, sino en la estructura académica real de la carrera de Computación, asegurando que cada herramienta y contenido desarrollado responda fielmente a las competencias que el estudiante realmente necesita adquirir. Al fundamentar el trabajo en los ejes temáticos clave, se garantiza que la tutoría sea una extensión coherente y valiosa del aprendizaje en el aula.

Se logró consolidar una arquitectura web y una estructura de datos sólidas, diseñadas específicamente para dar soporte a la gestión independiente de las tutorías. Este proceso de planificación técnica permitió crear un entorno digital organizado donde la información fluye con claridad, asegurando que el sistema sea capaz de responder a las exigencias académicas de la carrera de Computación. Al priorizar una base tecnológica bien estructurada, se garantiza que el estudiante cuente con una plataforma estable y eficiente que facilite su propio proceso de aprendizaje de manera autónoma.

Se logró materializar el diseño técnico en un prototipo funcional de tutor inteligente, diseñado específicamente para acompañar el aprendizaje en la carrera de Computación. Este desarrollo permitió integrar la lógica de las tutorías académicas en una herramienta interactiva que facilita la gestión autónoma del conocimiento. Al centrar el proceso en las necesidades del estudiante, se obtuvo una plataforma práctica que transforma la teoría en una experiencia de apoyo constante, asegurando que el recurso sea tanto útil como accesible para el usuario final.

Se validó el prototipo del tutor inteligente mediante pruebas funcionales y de usabilidad con estudiantes de la Facultad de Computación, lo que permitió confirmar que la herramienta es intuitiva y responde eficazmente a las necesidades

reales de aprendizaje. Este proceso de verificación directa con los usuarios fue fundamental para asegurar que el sistema no solo sea técnicamente robusto, sino también un apoyo práctico y confiable que facilita una interacción fluida y aporta un valor significativo a la formación académica en la carrera.

Recomendaciones

Para garantizar la continuidad, sostenibilidad y evolución del prototipo desarrollado, se realizan las siguientes recomendaciones:

Se recomienda que este prototipo y su documentación técnica sirvan como punto de partida para futuras investigaciones. La arquitectura modular del sistema permite a los nuevos estudiantes de tesis desarrollar módulos complementarios, como la integración de reconocimiento de voz, versiones para dispositivos móviles o la expansión a otras materias complejas.

Para maximizar el impacto de la herramienta, se recomienda al director del programa que evalúe la integración del prototipo con los sistemas oficiales de la UCSG como el Moodle o el Sistema de Gestión Académica. Esto facilitaría el inicio de sesión único para los usuarios y permitiría que las estadísticas de uso de los tutores se reflejen directamente en los indicadores de rendimiento institucional.

Es fundamental promover sesiones de socialización con el profesorado de la facultad para familiarizarlo con el Panel de Gestión. Al demostrar cómo la herramienta automatiza sus informes de tutoría, se fomentará su adopción y se motivará al profesorado a recomendar el uso del tutor inteligente a sus estudiantes como recurso oficial del curso.

Referencias Bibliográficas

- Ali Alkhatlan, J. K. (23 de Diciembre de 2018). *arxiv.org*. Obtenido de Intelligent Tutoring Systems: A Comprehensive Historical Survey with Recent Developments: <https://arxiv.org/abs/1812.09628>
- Azure DevOps. Microsoft Learn. (s.f.). *¿Qué es Git?* Obtenido de <https://www.scribbr.es/citar/generador/folders/6oNpfpIBqO232A3qBJdcEb/lists/69DDeN0rtMR028Zywn0dLq/>
- Casado, P. F. (2023). *Construcción y diseño de páginas web con HTML, CSS y JavaScript*. RaMa.
- Chávez, M. H. (9 de Enero de 2025). *RIDE Revista Iberoamericana para la investigación y el desarrollo Educativo*. Obtenido de RIDE Revista Iberoamericana para la investigación y el desarrollo Educativo: <https://www.ride.org.mx/index.php/RIDE/article/view/848>
- Ciencia Latina Revista Científica Multidisciplinar. (s.f.). *Integración de la Inteligencia Artificial en la Educación*.
- FastAPI. (s.f.). *FastAPI*. Obtenido de <https://fastapi.tiangolo.com/es/>
- Hernández-Sampieri, R. &. (2022). *Metodología de la Investigación*. Mc Graw Hill Education.
- Herrador, S. (s.f.). *HTML & CSS Fácil y sencillo*. Lulu.com.
- Microsoft Azure. (s.f.). *Microsoft Azure*. Obtenido de <https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-azure>
- Pino, R. R. (2022). Editorial Feijóo.
- PostgreSQL. (2025). *PostgreSQL*. Obtenido de <https://www.postgresql.org/about/>
- Rodrigo Elias Francisco, F. d. (2022). *Intelligent Tutoring System for Computer Science Education and the Use of Artificial Intelligence: A Literature Review*. Uberlandia: Faculty of Computer, Federal University of Uberlândia. Obtenido de <https://share.google/eBgDm1QaJLm35cS3n>
- Sophia Yang, Z. W. (8 de Junio de 2021). *Analyzing Patterns in Student SQL Solutions via Levenshtein Edit Distance*. Obtenido de [dl.acm.org: https://dl.acm.org/doi/10.1145/3430895.3460979](https://dl.acm.org/doi/10.1145/3430895.3460979)
- Sucari, B. C. (2024). *HERRAMIENTAS DIGITALES EDUCATIVAS*. Instituto Universitario de Innovación Ciencia y Tecnología Inudi Perú S.A.C.

Anexos

Ilustración 16 Encuesta de Distribución de la población estudiantil por semestre.

Encuesta de Estudio

Quando envíe este formulario, no recopilará automáticamente sus detalles, como el nombre y la dirección de correo electrónico, a menos que lo proporcione usted mismo.

* Obligatorio

Acerca de usted

Su información es extremadamente confidencial, la siguiente información es solo para fines de investigación internos, toda la información no se compartirá externamente.

1. ¿En qué semestre se encuentra? *

- Primer semestre
- Segundo semestre
- Tercer semestre
- Cuarto semestre
- Quinto semestre o superior

2. ¿Ha cursado anteriormente la asignatura de POO? *

- Sí
- No
- Me encuentro cursando

3. ¿Cómo califica su nivel actual en POO? *

- Alto
- Medio
- Bajo

Ilustración 17 Encuesta Autopercepción del nivel de conocimiento en Programación Orientada a Objetos.

Encuesta de Estudio

* Obligatorio

Tutorías académicas actuales

4. Esta sección tiene como objetivo identificar la percepción de los estudiantes sobre las tutorías académicas actuales en la asignatura de Programación Orientada a Objetos. *

	Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo
La asignatura de Programación Orientada a Objetos presenta un alto nivel de dificultad.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
He requerido tutorías académicas para comprender los contenidos de la asignatura.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Las tutorías académicas actuales cubren mis necesidades de aprendizaje.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
La disponibilidad de horarios para tutorías académicas es adecuada.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Las tutorías académicas contribuyen a mejorar mi rendimiento académico.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Atrás Siguiente

Ilustración 18 Encuesta sobre el Uso de tecnologías y aprendizaje autónomo.

Encuesta de Estudio
🔍 ...

Uso de tecnología y aprendizaje autónomo
🔍

5. Esta sección busca conocer el nivel de uso de herramientas tecnológicas por parte de los estudiantes y su percepción sobre el aprendizaje autónomo como apoyo para la comprensión de los contenidos de la asignatura de Programación Orientada a Objetos. 🔍

	Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo
Me siento capaz de aprender Programación Orientada a Objetos de forma autónoma con apoyo tecnológico.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Utilizo herramientas digitales (videos, plataformas, software) para reforzar mis conocimientos en programación.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Considero que el uso de tecnología facilita el aprendizaje de programación.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Atrás
Siguiente

Ilustración 19 Encuesta demanda y necesidad de tutorías académicas.

Encuesta de Estudio
🔍 ...

Tutor Inteligente
🔍

6. Esta sección tiene como finalidad recopilar la opinión de los estudiantes sobre la utilidad y aceptación de un Tutor Inteligente como herramienta de apoyo en la asignatura de Programación Orientada a Objetos. 🔍

	Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo
Un Tutor Inteligente disponible en cualquier momento sería útil para la asignatura de Programación Orientada a Objetos.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Me gustaría recibir tutorías personalizadas según mi nivel de conocimiento.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Estaría dispuesto(a) a utilizar un Tutor Inteligente como apoyo en la asignatura.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Atrás
Siguiente



DECLARACIÓN Y AUTORIZACIÓN

Yo, **Páliz Sánchez, Rodolfo David** con C.C: # **092965449-9** autor/a del proyecto de tecnología de información, **Prototipo de tutor inteligente para la gestión autónoma de tutorías académicas en la asignatura de programación orientada a objetos en la carrera de computación de la Universidad Católica de Santiago de Guayaquil**, requerido para la obtención del título de **Ingeniero en Ciencias de la Computación** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 4 de marzo del 2026.

Nombre: **Páliz Sánchez, Rodolfo David**

C.C: **0929654499**

REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

TEMA Y SUBTEMA:	Prototipo de tutor inteligente para la gestión autónoma de tutorías académicas en la asignatura de programación orientada a objetos en la carrera de computación de la Universidad Católica de Santiago de Guayaquil.		
AUTOR(ES)	Páliz Sánchez, Rodolfo David		
REVISOR(ES)/TUTOR(ES)	Castro Aguilar, Gilberto Fernando		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil		
FACULTAD:	Ingeniería		
CARRERA:	Computación		
TÍTULO OBTENIDO:	Ingeniero en Ciencias de la Computación		
FECHA DE PUBLICACIÓN:	4 de marzo del 2026	No. DE PÁGINAS:	40 p.
ÁREAS TEMÁTICAS:	Educación superior, Proceso de aprendizaje, Tecnología educativa, Programación informática, Innovación pedagógica, Aprendizaje autónomo.		
PALABRAS CLAVES/ KEYWORDS:	Programación Orientada a Objetos, abstracción, aprendizaje, herramienta didáctica.		
RESUMEN/ABSTRACT:	Durante la carrera de Ingeniería en ciencias de la computación y sus respectivas asignaturas como las tutorías académicas son un elemento fundamental que además de servir como una calificación en el porcentaje académico de aprobación sirve enormemente al aporte de conocimiento y aprendizaje de los estudiantes; agregado a esto la materia de programación orientada a objetos (POO) es un eje primordial en la formación de los estudiantes, sin embargo, el nivel de abstracción y complejidad en cada uno de sus conceptos que posee la materia provoca la generación de aquellas dificultades continuas en el aprendizaje de los estudiantes. En el presente trabajo de integración curricular se propone el desarrollo de una herramienta interactiva que tiene como finalidad poder fortalecer la autonomía de los estudiantes, facilitar la comprensión a conceptos referentes de la materia de POO y la consolidación de información relevante de la asignatura.		
ADJUNTO PDF:	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
CONTACTO CON AUTOR/ES:	Teléfono: +593-990251521 E-mail: Davispaliz@gmail.com – Rodolfo.Paliz@cu.ucsg.edu.ec		
CONTACTO CON LA INSTITUCIÓN (COORDINADOR DEL PROCESO UTE):	Toala Quimí, Edison José Teléfono: +593-990-976776 E-mail: edison.toala@cu.ucsg.edu.ec		
SECCIÓN PARA USO DE BIBLIOTECA			
Nº. DE REGISTRO (en base a datos):			
Nº. DE CLASIFICACIÓN:			
DIRECCIÓN URL (tesis en la web):			