

Caso de Uso: Comparación entre Entregas de Estudiantes
con IA GEMINI
Versión:2.2

CodeGuardGT
Caso de Uso: Comparación entre Entregas de Estudiantes con IA GEMINI

Caso de Uso: Comparación entre Entregas de Estudiantes con IA GEMINI

1. Comparación entre Entregas de Estudiantes con IA GEMINI

1.1 Breve descripción del caso de uso

Este caso de uso describe cómo un docente utiliza el sistema para revisar evaluaciones estudiantiles, enfocándose en detectar plagio en códigos SQL entregados. El docente accede a las evaluaciones, analiza similitudes en los documentos SQL y genera un informe de plagio.

2. Definiciones, Acrónimos, y Abreviaturas

- **CodeGuardGT:** Aplicativo para la detección de plagio en código SQL.
- **SQLAlchemy:** Biblioteca de Python.
- **Selenium:** Herramienta de automatización de navegadores web.
- **Caso de uso:** Descripción de cómo los usuarios interactúan con el sistema para lograr un objetivo.
- **MODAL:** mensaje emergente que retorna la aplicación, sea favorable o errónea.
- **Contraseña:** Palabra “Contraseña” en el aplicativo.
- **Api:** Una API (Interfaz de Programación de Aplicaciones) es un conjunto de reglas y protocolos que permite que diferentes aplicaciones se comuniquen entre sí, facilitando el intercambio de datos y funciones.
- **Botones del sistema:** Entre comillas.
- **Mensajes del sistema:** En negrita.
- **Mensaje satisfactorio:** Verde.
- **Mensaje de error:** Rojo.

3. Flujo de Eventos

3.1 Flujo Básico de Comparación entre Entregas de Estudiantes con IA GEMINI

ACTOR	SISTEMA
4. El “Usuario Docente” ingresa al sistema.	5. El sistema carga la página de LOGIN cargando un formulario donde solicitará DNI y contraseña.
6. El “Usuario Docente” ingresa sus credenciales.	
7. El “Usuario Docente” hace clic en el botón “Iniciar Sesión”.	8. El sistema verifica las credenciales ingresadas.
	9. El sistema redirecciona a la pantalla principal del “Usuario Docente”.

CodeGuardGT
Caso de Uso: Comparación entre Entregas de Estudiantes con IA GEMINI

10. El “Usuario Docente” desglosa el menú lateral izquierdo y selecciona el apartado “Gestión de Evaluaciones”	11. El sistema muestra la pantalla de “Gestión de Evaluaciones” presenta una cartilla principal de Evaluaciones por Curso donde se muestran los cursos y sus respectivas evaluaciones relacionadas con el docente.
12. El “Usuario Docente” hace clic en el botón “Realizar Revisión” de la evaluación que desea revisar	13. El sistema presenta un modal que permitirá seleccionar uno de los diferentes métodos de revisión de plagio y un amplificador de umbral para medir de 0.0 a 1.0 el porcentaje de evaluación de plagio y un botón para confirmar.
	13. El sistema muestra un modal con en el cual el “Usuario Docente” interactúa con un TextBox para ingresar sus parámetros en el prompt de la IA.
	14. El sistema realiza una consulta interna accediendo a la IA de Google(GEMINI)
	15. El sistema consume la API de GEMINI, mediante los parámetros hará la detección de plagio, enviando los documentos obtenidos para analizar las similitudes entre ellos.
	16. El sistema muestra un modal con el mensaje Resultado de la Revisión y los detalles del análisis de plagio.
17. El “Usuario Docente” visualiza y hace clic en el botón de “Cerrar” luego de visualizar el contenido del modal Resultado de la Revisión.	

CodeGuardGT
Caso de Uso: Comparación entre Entregas de Estudiantes con IA GEMINI

3.2 Flujos Alternativos

3.3 Flujo alternativo 1: *Enlace del estudiante duplicado (paso 15)*

ACTOR	SISTEMA
	1. Desde el punto 15 del flujo básico.
	2. El sistema muestra un modal con el mensaje Resultado de la Revisión y los detalles del análisis de plagio. y agrega un apartado que indique el enlace duplicado en la entrega, presentando cualquier plagio detectado.
	3. Regresa al punto 17 del flujo básico.

3.4 Flujo alternativo 2: *Una entrega o no hay entregas para revisión (paso 15)*

ACTOR	SISTEMA
	1. Desde el punto 15 del flujo básico.
2. El “Usuario Docente” hace clic en el botón “Realizar Revisión” de la evaluación que desea revisar	3. El sistema presenta un aviso de error Error en la solicitud: error - INTERNAL SERVER ERROR
	4. Regresa al punto 11 del flujo básico.

Requisitos Especiales

- **Rendimiento:** CodeGuardGT debe ser capaz de comparar hasta 20 entregas de código SQL en menos de 5 minutos para garantizar la eficiencia del proceso de revisión.
- **Seguridad:** El análisis debe realizarse garantizando la confidencialidad e integridad de los datos, evitando accesos no autorizados o modificaciones a las entregas de los estudiantes.
- **Conexión a Telegram:** El sistema debe establecer una conexión segura con la API de Telegram para enviar notificaciones sin interrupciones, asegurando la correcta entrega del reporte de evaluación.
- **Integridad de Datos:** Las comparaciones de plagio y las consultas a la base de datos deben realizarse sin comprometer la integridad de los datos de los estudiantes ni la confidencialidad de la información.

Precondiciones

- Las entregas de los estudiantes deben estar completas y accesibles en la plataforma donde se gestionan (por ejemplo, un LMS o base de datos).
- CodeGuardGT debe contar con acceso a dichas entregas para realizar el análisis.
- El “Usuario Docente” debe haber iniciado sesión con credenciales válidas.
- El sistema debe tener acceso a las APIs correspondientes, tanto la API interna para la detección de plagio como la API externa de Telegram para notificaciones.
- El sistema debe estar correctamente configurado para realizar consultas a las bases de datos (tablas curso, evaluación, documentos) mediante SQLAlchemy y acceso a Selenium para la extracción de datos.
- El “Usuario Docente” debe tener una conexión a internet estable (se recomienda tener al menos una velocidad de internet de 1 Mbps; sin embargo, para una experiencia más fluida y sin interrupciones, una velocidad de 3 a 5 Mbps sería lo ideal) para acceder al aplicativo web.

Postcondiciones

- Cualquier entrega que haya sido omitida por problemas técnicos o de acceso es claramente indicada en el informe, junto con la razón de la omisión.
- Un informe detallado sobre la similitud entre los documentos SQL entregados por los estudiantes debe ser generado y accesible para el “Usuario Docente”, ya sea por la plataforma o por un canal alternativo (correo, descarga directa).
- Cualquier error en el proceso de generación del informe o el envío de la notificación debe ser registrado en el aplicativo web, y el “Usuario Docente” debe ser informado con opciones para reintentar.