
Universidad La Salle

CodeGuardGT
Plan de Iteración 1

CodeGuardGT
Plan de Iteración

Plan de Iteración 1

1. Introducción

1.1 Alcance

Este documento detalla el plan de la Iteración 1 del proyecto CodeGuardGT, un sistema de detección de plagio enfocado en el análisis de código SQL desde el compilador online SQLFiddle. El plan cubre los casos de uso a desarrollar, los recursos, las actividades y los entregables necesarios para completar esta primera **Iteración 1**, el proyecto **CodeGuardGT** implementará las funcionalidades esenciales para la detección de plagio en SQL y se realizará una prueba interna con los casos de uso seleccionados. También se corregirán defectos encontrados y se implementarán cambios sugeridos durante la revisión interna del equipo.

1.2 Definiciones, acrónimos, y abreviaturas

- **CodeGuardGT:** Aplicativo para la detección de plagio en código SQL.
- **SQLFiddle:** Herramienta online para ejecutar y compartir consultas SQL.
- **Iteración:** Ciclo de desarrollo en el que se implementan, prueban y mejoran funcionalidades específicas del sistema.
- **Caso de uso:** Descripción de cómo los usuarios interactúan con el sistema para lograr un objetivo.

2. Casos de Uso

El listado de casos de uso a implementar y probar en esta iteración es el siguiente:

CU Entrega Evaluación Estudiante

CU Caso de Uso RegistrarUsuario

CU CambiarContraseñaPredeterminada

CU CambiarContraseñaAdministrador

CU Comparación entre Entregas de Estudiantes

- Luego de haber culminado la evaluación, el sistema web realizará la captura y comparación de los códigos de todos los alumnos uno a uno.

3. Recursos

Persona	Roles
Pedro Humberto Rondon Ponce	Líder de proyecto
Jose Carlos Peraltilla Nuñez	Revisor de contenido - pruebas
Christian Omar Rodriguez Huamanñahui	Gestor Calidad - Experto Técnico
Roger Euclides Infa Sanchez	Desarrollador Full Stack
Carla Fernanda Ropa Calizaya	Arquitecto de software
Melany Yasmin Lazo Arana	Arquitecto de software
Carlos Daniel Corrales Yarasca	Gestor de Calidad - Experto de Negocio

Entidades de Data	Descripción
<ul style="list-style-type: none"> • Usuarios 	<ul style="list-style-type: none"> • Almacena los datos de los usuarios, como dni, nombres, apellidos, correo, fecha nacimiento y contraseña.

CodeGuardGT
Plan de Iteración

<ul style="list-style-type: none"> • Administración • Docentes • Estudiantes • Documentos • Notas • Incidencias • Plagios 	<ul style="list-style-type: none"> • Representa a los usuarios del área de administración, vinculados por dni_usuario a la entidad de usuarios. • Representa a los usuarios docentes, vinculados por dni_usuario a la entidad de usuarios. • Representa a los usuarios estudiantes, vinculados por dni_usuario a la entidad de usuarios. • Registra los documentos subidos por los estudiantes, con atributos como url_documento, fecha_subida y estado_analisis. • Almacena las notas de los estudiantes, con id_nota, dni_estudiante, nota y fecha_registro. • Registra problemas relacionados con los documentos subidos, con detalles como incidencia, descripción, y fecha_incidencia. • Almacena información sobre el porcentaje de plagio en los documentos, fecha_analisis, detalles_plagio y estado.
Fuente del Plagio	<p>Se maneja información de plagio a partir de código SQL que los usuarios suben a través de SQLFiddle. Las principales entidades de datos del proyecto son:</p> <ul style="list-style-type: none"> • Usuarios: Datos del usuario registrado en CodeGuardGT. • Archivos SQL: Código SQL subido para su análisis. • Historial de análisis: Registro de las evaluaciones de plagio con detalles de similitud.

Otros (hardware, ambientes)	Descripción
Entorno de pruebas:	El cliente podrá acceder a un ambiente de pruebas en línea donde podrá interactuar con las funcionalidades de CodeGuardGT. Las pruebas iniciales se realizarán en un entorno de desarrollo local y luego se desplegarán en un servidor de pruebas.
Herramientas del cliente:	Se proporcionará acceso a la herramienta de pruebas SQLFiddle para que los usuarios puedan cargar el código SQL y someterlo a evaluación.

4. Plan

Actividad	Fecha de culminación
Implementación de casos de uso	04/09/24
Realización de pruebas internas	11/09/24
Corrección de defectos e implementación de	18/09/24

CodeGuardGT
Plan de Iteración

cambios	
Ciclo 1: Corrección de defectos / Implementación de cambios	Realizar pruebas internas, correcciones, puede ser un ciclo semanal por términos de tiempo, en cada ciclo después de las pruebas y se van levantando requisitos, se corrige, corrección, defecto, implementación, cambios, pruebas de lunes a viernes, programación, sábado domingo, pruebas y lunes chequeo, martes a viernes cambios
Ciclo 1: Corrección de Defectos / Implementación de Cambios	<p>Este ciclo consistirá en la identificación y corrección de defectos encontrados durante las pruebas internas. El ciclo se repetirá de manera semanal con la siguiente estructura:</p> <ul style="list-style-type: none"> • Lunes a viernes: Desarrollo, pruebas y verificación de defectos. • Sábado y domingo: Pruebas internas y ajustes finales. • Lunes: Chequeo general y evaluación de cambios. • Martes a viernes: Implementación de comprobaciones sugeridas.

5. Entregables

Artefactos que programas y servidor a nivel cliente
documentación y programas
Artefactos programados: Implementación de los casos de uso en CodeGuardGT.
Documentación: Descripción técnica de los casos de uso implementados y manuales de usuario.
Servidor cliente: Código desplegado para el entorno de pruebas accesible para el cliente.

6. Criterios de Evaluación

Para considerar exitosa la Iteración 1, se deben cumplir los siguientes criterios:

1. Cumplimiento de pruebas de sistema: El sistema debe pasar todas las pruebas de sistema realizadas para los casos de uso implementados.
2. Cumplimiento de pruebas unitarias: Las pruebas unitarias deben estar correctamente implementadas y ejecutadas desde una suite de pruebas.
3. Revisiones externas: Se debe pasar la revisión por pares realizada por equipos externos.

7. Supuestos y restricciones

CodeGuardGT
Plan de Iteración

Supuestos: Se asume que SQLFiddle estará disponible y en funcionamiento continuo durante el desarrollo de la iteración. También se considera que los usuarios ya cuentan con acceso al sistema.

Restricciones: El sistema de detección de plagio dependerá del buen funcionamiento de SQLFiddle. No se puede garantizar el análisis en caso de caídas o problemas técnicos en dicha herramienta. Cualquier limitación de terceros o licencias de software que no estén bajo nuestro control será documentada.