



**PROYECTO FINAL
SISTEMA DE GESTIÓN DE PRUEBAS DE SOFTWARE**

INTEGRANTES:

**ALDEMAR VIVAS AMARILLO - 1968062
ANA ALEJANDRA BOLAÑOS - 1968026
HAROLD MONTANO - 1968067
JUAN STEBAN VELOZA - 1968025**

DOCENTE:

ING. JUAN PABLOS PINILLOS REINA

PRUEBAS DE SOFTWARE

**UNIVERSIDAD DEL VALLE – SEDE TULUÁ
FACULTAD DE INGENIERÍA
INGENIERÍA DE SISTEMAS
2023**

TABLA DE CONTENIDO

Informe.....	3
Arquitectura del sistema.....	5
Diagramas de diseño.....	6
Informe de pruebas.....	10
Conclusiones.....	12
Referencias.....	13

INFORME

- **Registro de Proyectos**

Crear Proyecto: Permite a los usuarios registrar nuevos proyectos proporcionando información detallada, como nombre del proyecto, descripción y miembros del equipo.

Gestionar Proyecto: Ofrece herramientas para la administración continua del proyecto, incluyendo la capacidad de editar detalles del proyecto, agregar o eliminar miembros del equipo, y asignar roles y responsabilidades a cada miembro.

- **Creación de casos de prueba**

Crear Caso de Prueba: Permite a los testers crear casos de prueba especificando información detallada, como título, descripción, pasos a seguir, datos de prueba y expectativas.

Documentar Caso de Prueba: Facilita la documentación exhaustiva de cada caso de prueba para una comprensión clara y una ejecución efectiva.

- **Planificación de Pruebas**

Asignar Caso de Prueba a Etapa: Permite a los testers asignar casos de prueba a diferentes etapas del desarrollo del software para asegurar una cobertura completa.

Establecer Prioridades y Fechas Límite: Facilita la planificación eficiente permitiendo establecer prioridades para la ejecución de pruebas y asignar fechas límite.

- **Ejecución de Pruebas**

Registrar Resultados de Pruebas: Permite a los testers registrar los resultados de las pruebas, indicando si un caso de prueba fue exitoso, falló o no se ejecutó.

Adjuntar Evidencia: Ofrece la posibilidad de adjuntar evidencia, como capturas de pantalla o registros, para respaldar los resultados de las pruebas.

- **Seguimiento de Errores**

Acceder al Sistema de Seguimiento de Errores: Integración con un sistema de seguimiento de errores que permite registrar y priorizar problemas identificados durante las pruebas.

Registrar Problema: Permite a los usuarios registrar problemas identificados durante las pruebas, proporcionando información detallada sobre el error.

Asignar Problema para Corrección: Facilita la asignación de errores a los miembros del equipo responsables de corregirlos.

- **Generación de Informes**

Crear Informe de Progreso de Pruebas: Permite a los administradores de proyecto generar informes detallados sobre el progreso de las pruebas.

Incluir Métricas de Calidad: Proporciona métricas de calidad para evaluar la eficacia de las pruebas.

Incluir Cobertura de Pruebas: Ofrece estadísticas sobre la cobertura de pruebas, identificando áreas del software que han sido probadas y aquellas que necesitan más atención.

Incluir Estadísticas de Errores: Proporciona estadísticas detalladas sobre la frecuencia y la gravedad de los errores identificados.

Arquitectura del Sistema

La arquitectura del sistema sigue un modelo de tres capas:

- **Capa de Presentación**

La interfaz de usuario (UI) proporciona una experiencia interactiva y amigable para los usuarios.

Se utiliza un marco de desarrollo basado en Python, para la implementación del sistema.

- **Capa de Lógica**

Contiene la lógica central del sistema, gestionando la creación, actualización y eliminación de proyectos, casos de prueba, y la asignación de roles y responsabilidades.

Implementa la lógica para la planificación de pruebas, ejecución de pruebas, seguimiento de errores y generación de informes.

- **Capa de Almacenamiento de Datos**

Utiliza una base de datos relacional PostgreSQL para almacenar la información del proyecto, casos de prueba, resultados de pruebas, y detalles de errores.

DIAGRAMAS DE DISEÑO

Diagrama de Clases Base de Datos:

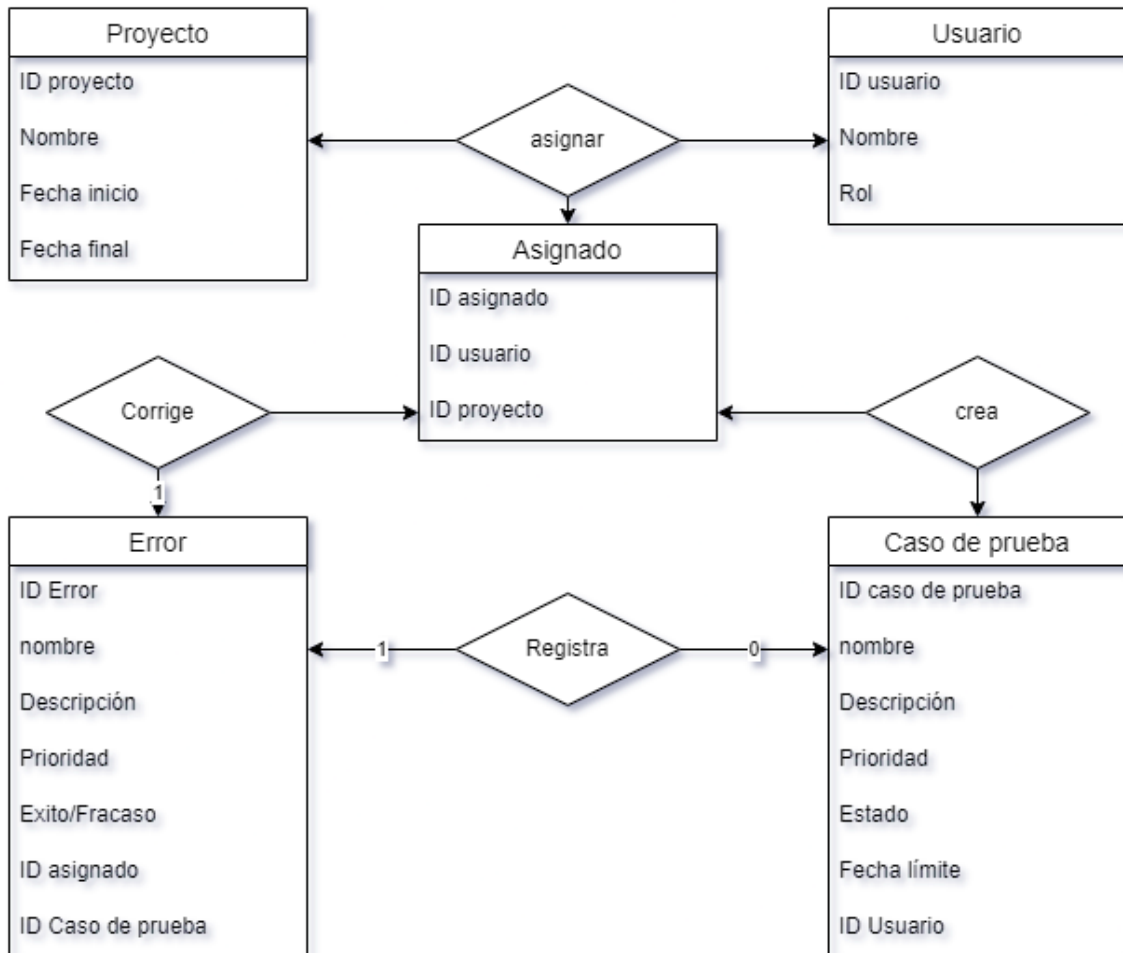
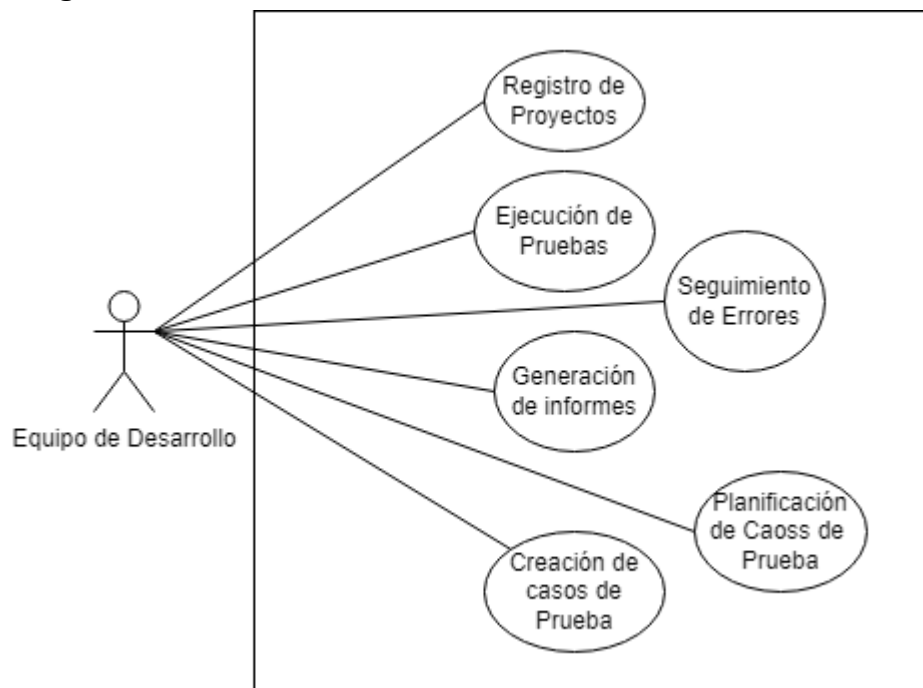
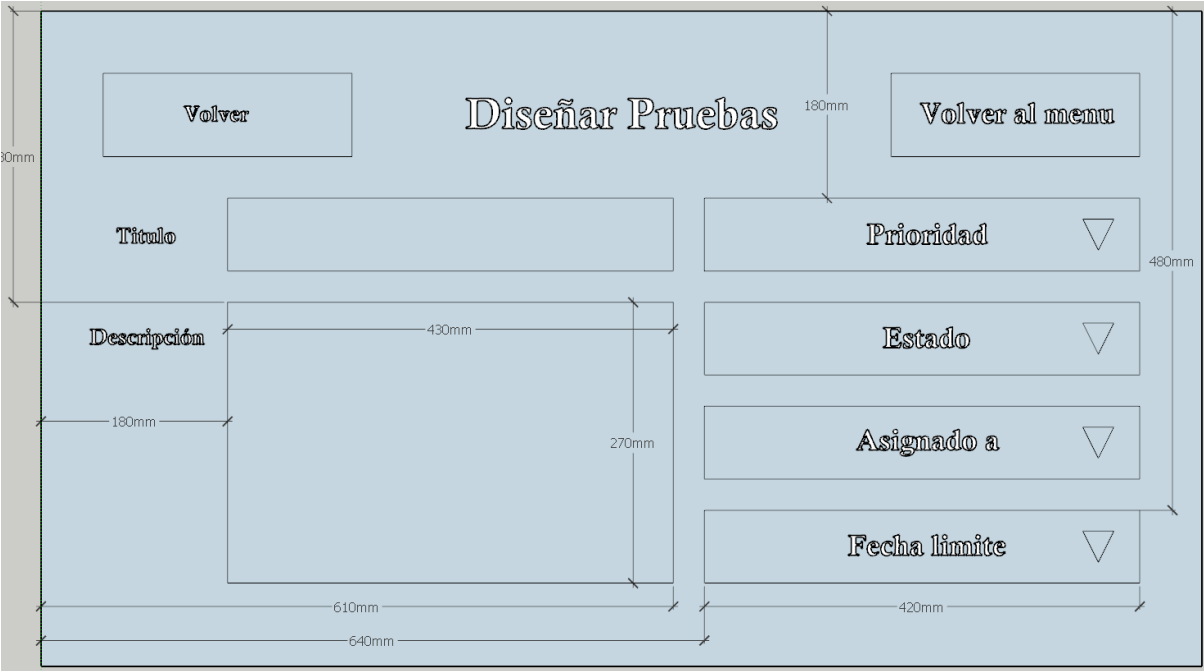
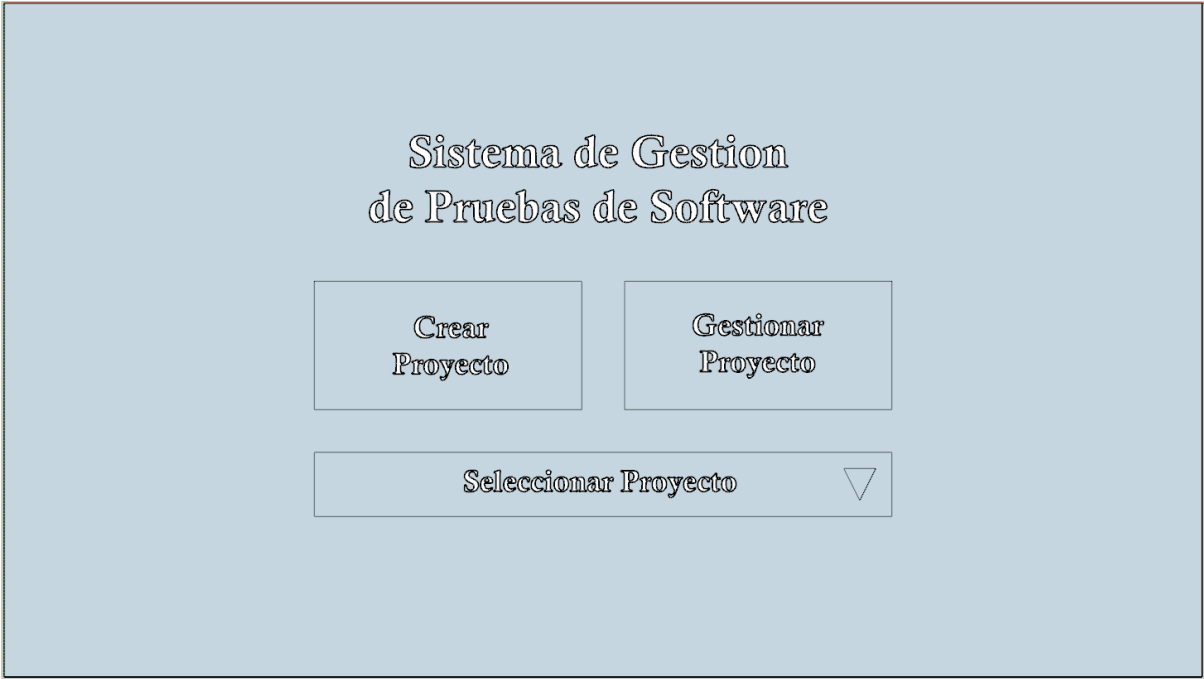


Diagrama de Casos de Uso:



Diagramas de Aplicación:



Volver

Gestion de errores

Volver al menu

Título		Severidad	▼
Descripción		Estado	▼
		Asignado a	▼
		Caso de prueba	▼

[illegible]

INFORME DE PRUEBAS



Sistema de gestion de pruebas de software

Volver

Diseñar Pruebas

Volver al Menú

Título

Prueba ejemplo

Descripción

Esta prueba busca verificar...

Baja

Estado

Asignar a

Elegir fecha

12/11/23

December

Mon	Tue	Wed	Thu
48	27	28	29
49	4	5	6
50	11	12	13

Sistema de gestion de pruebas de software

Agregar

Editar

Opcion a gestionar

Volver al Menú

Estado	Prioridad	Designado a	Título de prueba
Listo	Mediana	Juan	Prueba unitaria
Detenido	Baja	Ana	Prueba de usabilidad
En curso	Alta	Harold	Prueba de aceptación

CONCLUSIONES

- **Eficiencia en la Gestión de Pruebas**

La aplicación proporciona una plataforma eficiente para la gestión de pruebas en proyectos de desarrollo de software, permitiendo a los usuarios planificar, ejecutar y hacer un seguimiento efectivo de las pruebas.

- **Colaboración Mejorada**

La asignación de roles y responsabilidades facilita la colaboración dentro del equipo, asegurando que cada miembro tenga tareas definidas y acceso a la información relevante.

- **Documentación Detallada de Casos de Prueba**

La capacidad de crear casos de prueba con descripciones, pasos a seguir, datos de prueba y expectativas facilita una documentación detallada, mejorando la comprensión y ejecución de las pruebas.

- **Planificación Efectiva**

La aplicación permite asignar casos de prueba a diferentes etapas del desarrollo, establecer prioridades y fechas límite, lo que contribuye a una planificación efectiva de las pruebas y garantiza una cobertura completa.

- **Seguimiento de Errores Integrado**

La integración de un sistema de seguimiento de errores mejora la eficiencia al registrar, priorizar y asignar problemas identificados durante las pruebas, facilitando su corrección por parte del equipo.

REFERENCIAS

- <https://www.istqb.org/>
- Clean Code: "Clean Code: A Handbook of Agile Software Craftsmanship" por Robert C. Martin.
- <https://peps.python.org/pep-0008/>
- "Foundations of Software Testing" de Dorothy Graham y Erik Van Veenendaal.
- "Agile Testing: A Practical Guide for Testers and Agile Teams" de Lisa Crispin y Janet Gregory.