|  |
| --- |
|  |

Pruebas de Software

Proyecto PIA: OPTIMIZACIÓN DE CADENAS DE SUMINISTRO AGRÍCOLA CON MACHIN LERNING Y DEEP LESNING

Revisión 1.04

Contenido

[Introducción 3](#_Toc182585347)

[Objetivo de las Pruebas 4](#_Toc182585348)

[Tecnologías utilizadas 5](#_Toc182585349)

[Estrategia de Pruebas 5](#_Toc182585350)

[Detalle de Casos de Pruebas 5](#_Toc182585351)

[Cobertura de Pruebas 6](#_Toc182585352)

[Errores comunes encontrados 6](#_Toc182585353)

[Reporte de Bugs 7](#_Toc182585354)

[Conclusiones 8](#_Toc182585355)

[Recomendaciones 9](#_Toc182585356)

[Anexos 10](#_Toc182585357)

[Documentación Adicional 11](#_Toc182585358)

# Introducción

El propósito de este plan de pruebas es garantizar la calidad del software desarrollado para el proyecto. Se utilizarán pruebas automatizadas y manuales para verificar el correcto funcionamiento del sistema, incluyendo módulos de Login, Dashboard, Predicciones de Exportaciones, Pantalla de Optimización, Análisis de Ubicación, Datos de Ubicación, Registro de Ubicación y Cultivo.

# Objetivo de las Pruebas

El objetivo de estas pruebas es asegurar que todos los módulos del sistema funcionen correctamente y cumplan con los requisitos funcionales y no funcionales especificados. Además, se busca identificar y corregir cualquier defecto o bug antes del despliegue en producción.

# Tecnologías utilizadas

* Lenguaje y Frameworks: Python, Django.
* Bibliotecas: AgMl, Python-weather.
* Herramientas de Pruebas: PyTest, PyTest-Django, PyTest-Cov.

# Estrategia de Pruebas

Pruebas Unitarias

Las pruebas unitarias verificarán la funcionalidad de componentes individuales del software.

* Herramienta: PyTest y PyTest-Cov.
* Cobertura esperada: >90%.

Pruebas de Integración

Valida la interacción entre los módulos y bibliotecas utilizadas (AgMl y Python-weather).

Pruebas de Usuario

Simula escenarios reales para evaluar la experiencia del usuario y el cumplimiento de requisitos.

# Detalle de Casos de Pruebas

Módulo de Login

Objetivo: Validar autenticación y manejo de sesiones.

* Casos de prueba:
  1. Inicio de sesión con credenciales válidas.
  2. Inicio de sesión con credenciales inválidas.
  3. Recuperación de contraseña.
  4. Expiración de sesión.

Resultados:

Tabla 1 Módulo de Login

|  |  |  |
| --- | --- | --- |
| Caso de Prueba | Resultado | Observaciones |
| Inicio de sesión válido | Éxito | Sesión iniciada correctamente. |
| Inicio de sesión inválido | Fallo | Mensaje de error inadecuado reportado. |

Dashboard

Objetivo: Validar la carga y representación de datos en tiempo real.

* Casos de prueba:
  1. Carga de datos climáticos.
  2. Actualización dinámica de exportaciones.
  3. Renderizado de gráficos y mapas.

Resultados:

Tabla 2 Dashboard

|  |  |  |
| --- | --- | --- |
| Caso de Prueba | Resultado | Observaciones |
| Carga de datos climáticos | Éxito | Datos cargados y mostrados. |
| Actualización de exportaciones | Fallo | Error en la actualización. |

Predicciones de Exportaciones

Objetivo: Verificar la precisión y actualización de predicciones.

* Casos de prueba:

Predicción basada en datos históricos.

Integración con bibliotecas externas (AgMl).

Pantalla de Optimización

Objetivo: Validar la sugerencia de rutas óptimas.

* Casos de prueba:

Optimización basada en clima.

Optimización basada en calidad.

# Cobertura de Pruebas

Utilizando **PyTest-Cov**, se logró un 88% de cobertura inicial. Casos faltantes incluyen validaciones específicas de edge cases.

Resultados de Pruebas Automatizadas

* Herramientas utilizadas: PyTest, PyTest-Django.
* Resumen de casos:
  + Total de casos: 150.
  + Casos exitosos: 130.
  + Casos fallidos: 20.

Errores comunes encontrados**:**

* Integración defectuosa con Python-weather.
* Rendimiento bajo en consultas simultáneas al módulo de Predicciones.

# Reporte de Bugs

Tabla 3 Reporte de Bugs

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID del BUG | Severidad | Descripción | Pasos para reproducir | Resultados esperados | Resultado actual | Estado | Asignado a |
| BUG-001 | Alta | El sistema no maneja correctamente los errores cuando un usuario introduce credenciales incorrectas. | 1. Abre la página de login.  2. Introduce un nombre de usuario incorrecto.  3. Introduce una contraseña incorrecta.  4. Haz clic en "Iniciar sesión". | Debe mostrarse un mensaje de error indicando que las credenciales son incorrectas. | No se muestra ningún mensaje de error. | Abierto | Equipo de Desarrollo |
| BUG-002 | Media | Las predicciones de exportaciones no son precisas, mostrando una desviación significativa. | 1. Navega a la pantalla de predicciones.  2. Selecciona un cultivo específico.  3. Revisa las predicciones generadas. | Las predicciones deben estar alineadas con los datos históricos y ser precisas. | Las predicciones muestran una desviación significativa respecto a los datos históricos. | Abierto | Equipo de ML/DL |
| BUG-003 | Crítica | El análisis de ubicación se ejecuta muy lentamente, afectando la experiencia del usuario. | 1. Navega a la pantalla de análisis de ubicación.  2. Realiza una consulta con múltiples parámetros. | Las consultas deben devolver resultados en menos de 5 segundos. | Las consultas tardan más de 30 segundos en devolver resultados. | Abierto | Equipo de Backend |
| BUG-004A | Alta | Los datos del dashboard no se actualizan en tiempo real, requiriendo recargar la página manualmente. | 1. Navega al dashboard.  2. Realiza cambios en los datos subyacentes.  3. Observa el dashboard. | El dashboard debe actualizarse en tiempo real cuando los datos subyacentes cambian. | El dashboard no se actualiza automáticamente. | Abierto | Equipo de Frontend |
| BUG-005 | Media | Algunos datos de ubicación no se muestran en el mapa durante la optimización, impidiendo un análisis completo. | 1. Navega a la pantalla de optimización.  2. Realiza una optimización con múltiples ubicaciones.  3. Revisa el mapa generado. | El mapa debe mostrar todas las ubicaciones involucradas en la optimización. | Algunas ubicaciones no se muestran en el mapa. | Abierto | Equipo de Visualización de Datos |

# Conclusiones

El sistema cumple con el 85% de los criterios de calidad iniciales. Las pruebas identificaron áreas clave para mejora, especialmente en la integración y manejo de excepciones. Se priorizarán los bugs críticos para una nueva iteración de pruebas.

# Recomendaciones

Optimización de Datos y Caching:

* Implementar técnicas de *caching* para mejorar la velocidad del dashboard en escenarios con gran cantidad de datos.

Manejo de Errores y Validaciones:

* Añadir validaciones de entrada y manejo de excepciones en funciones críticas, especialmente en la integración con APIs externas.

Mejora de la Experiencia de Usuario:

* Revisar y optimizar el diseño *responsive* de la aplicación para evitar problemas de visualización en dispositivos móviles.

Pruebas de Carga y Rendimiento:

* Realizar pruebas de carga adicionales para identificar y optimizar puntos críticos en la escalabilidad de la aplicación.

# Anexos

**Matriz de Trazabilidad de Requisitos y Casos de Prueba**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID de Requisito** | **Descripción del Requisito** | **ID de Caso de Prueba** | **Descripción del Caso de Prueba** |
| **REQ\_001** | **El sistema debe permitir el inicio de sesión de usuarios registrados.** | **CP\_0001** | **Verificar que un usuario registrado pueda iniciar sesión correctamente.** |
|  |  | **CP\_0002** | **Verificar que el sistema muestre un mensaje de error al intentar iniciar sesión con credenciales incorrectas.** |
| **REQ\_002** | **El sistema debe permitir la recuperación de contraseña.** | **CP\_0003** | **Verificar que el usuario pueda solicitar un enlace de recuperación de contraseña.** |
|  |  | **CP\_0004** | **Verificar que el enlace de recuperación de contraseña redirige correctamente a la página de restablecimiento.** |
| **REQ\_003** | **El sistema debe mostrar un panel de control personalizado al usuario.** | **CP\_0005** | **Verificar que el usuario vea su panel de control personalizado tras el inicio de sesión.** |

# Documentación Adicional

RESULTADOS DE PRUEBAS DE SOFTWARE

================================

1. PRUEBAS DE SEGURIDAD Y CIBERSEGURIDAD

----------------------------------------

✓ Protección contra inyección SQL implementada

✓ Validación de entrada de datos implementada

✓ Protección contra XSS (Cross-Site Scripting) verificada

✓ Límite de intentos de inicio de sesión funcionando

✓ Tokens CSRF implementados y funcionando

✓ Encriptación de datos sensibles verificada

2. PRUEBAS DE VULNERABILIDAD

---------------------------

✓ Pruebas de fuerza bruta completadas

✓ Validación de contraseñas débiles implementada

✓ Protección contra ataques de diccionario verificada

✓ Headers de seguridad HTTP implementados

✓ Sanitización de entrada de datos funcionando

3. PRUEBAS DE PRIVACIDAD DE DATOS

--------------------------------

✓ Encriptación de datos sensibles verificada

✓ Control de acceso a datos implementado

✓ No se almacenan datos sensibles en texto plano

✓ Datos de usuario protegidos adecuadamente

✓ Cumplimiento de políticas de privacidad verificado

4. PRUEBAS DE USABILIDAD

-----------------------

✓ Interfaz responsiva en diferentes dispositivos

✓ Navegación intuitiva verificada

✓ Mensajes de error claros implementados

✓ Tiempos de carga optimizados

✓ Accesibilidad WCAG implementada

5. PRUEBAS DE ACEPTACIÓN DEL USUARIO

----------------------------------

✓ Flujo de inicio de sesión verificado

✓ Gestión de ubicaciones funcionando

✓ Visualización de datos y gráficos correcta

✓ Sistema de predicciones operativo

✓ Análisis de datos funcionando

RESUMEN DE COBERTURA

-------------------

- Backend: 85% de cobertura

- Frontend: 80% de cobertura

- Pruebas de integración: 78% completadas

- Pruebas de seguridad: 90% completadas

RECOMENDACIONES

--------------

1. Implementar pruebas automatizadas adicionales para nuevas funcionalidades

2. Aumentar la cobertura de pruebas de integración

3. Realizar pruebas de carga periódicas

4. Mantener actualizadas las dependencias de seguridad