**Capstone “Malt technology”**

**Plan de pruebas**

**línea horizontal**

**Integrantes: Docente: Cindy Contador**

**Humberto Letelier Asignatura: Capstone**

**Álvaro Farias**  **Sección: 009D**

**Sebastián Brenet**



|  |  |  |
| --- | --- | --- |
| **VERSIÓN** |  |  |
| 1.0 |  |  |
| **RAZÓN** | | |
| 1.0 Creación del documento | | |

## Índice

[**Índice 2**](#_heading=h.i4pmq8vl1068)

[**Introducción 3**](#_heading=h.iq8jqb785jl)

[**Estrategia de Pruebas 4**](#_heading=h.uy31wplvg3sj)

[**Elementos a ser Probados 4**](#_heading=h.5loi422oa57j)

[**Criterios de Aceptación 5**](#_heading=h.4jjh7m88oey)

[**Planificación 6**](#_heading=h.xryjrc5kior3)

[**Entregables 6**](#_heading=h.xln2s15c0xph)

[**Riesgos 6**](#_heading=h.gb8ihiscuel1)



# **Introducción**

**Propósito**

El propósito de este plan de pruebas es garantizar que la aplicación móvil que estamos desarrollando para Maltexco, junto con su backend y base de datos, cumpla con todas las especificaciones establecidas. Además, busca identificar y corregir cualquier defecto antes de su implementación en el entorno de producción.

**Alcance**

Este plan abarca pruebas funcionales, de integración, de rendimiento, de seguridad y de usabilidad. Se evaluarán las principales funcionalidades de la aplicación móvil, la comunicación con el backend y la interacción con la base de datos, asegurando que cada componente funcione correctamente y de manera integrada.

**Objetivos**

Verificar que la comunicación entre la aplicación móvil, el backend y la base de datos sea estable, eficiente y segura.

Detectar y corregir errores en la interfaz de usuario, asegurando que la experiencia sea intuitiva y fluida.

Evaluar el rendimiento de la aplicación bajo diferentes condiciones de carga y uso.

Comprobar que todas las funcionalidades clave cumplen con los requisitos especificados.

Validar la seguridad del sistema, garantizando que los datos sensibles estén protegidos.

Asegurar la usabilidad, confirmando que el sistema es fácil de usar y accesible para los operarios de Maltexco.



# 

# Estrategia de Pruebas

**Pruebas Funcionales**

* + Validar que todas las funcionalidades de la aplicación móvil se comportan de acuerdo con los requisitos establecidos.

**Pruebas de Integración**

* + Verificar que la interacción entre el frontend (aplicación móvil) y el backend (API) se realice de manera correcta y sin interrupciones.

**Pruebas de Rendimiento**

* + Evaluar el desempeño de la aplicación bajo diferentes condiciones de carga, como múltiples usuarios simultáneos, para asegurar que responde eficientemente.

**Pruebas de Seguridad**

* + Probar los mecanismos de autenticación y autorización para asegurar que los datos transmitidos y almacenados estén cifrados adecuadamente.

**Pruebas de Usabilidad**

* + Evaluar la interfaz de usuario para garantizar que sea intuitiva, accesible y fácil de utilizar por los operarios de Maltexco, mejorando la experiencia del usuario.

# 

# **Elementos a ser Probados**

**Módulo de aplicación móvil**

* **Visualización de datos en tiempo real**: Validar que la aplicación muestra los datos correctos de temperatura, humedad y otros parámetros de las máquinas.
* **Notificaciones**: Probar las alertas en caso de que los parámetros excedan los límites predefinidos.
* **Input manual**: Asegurar que los operarios puedan ingresar datos adicionales sin problemas.

**Módulo de backend**

* **Extracción y procesamiento de datos**: Verificar que el backend esté extrayendo correctamente los datos de las máquinas, limpiándolos y alimentándose en la base de datos.
* **API**: Validar que la API REST expuesta funcione correctamente y responda adecuadamente a las solicitudes de la aplicación móvil.

**Base de datos**

* **Almacenamiento de datos**: Asegurar que los datos de las máquinas se almacenen correctamente y que puedan ser consultados de manera eficiente.
* **Consultas históricas**: Validar que se puedan realizar consultas históricas de los parámetros de las máquinas.

# 

# **Criterios de Aceptación**

**Criterios Funcionales**

* Todas las funcionalidades principales (visualización de datos, alertas, input manual) deben estar completamente operativas.
* El sistema debe ser capaz de manejar la carga esperada de usuarios simultáneos sin caída de rendimiento.
* La comunicación entre la aplicación móvil y el backend debe ser estable.

**Criterios de Rendimiento**

* La aplicación debe poder manejar usuarios simultáneos sin tiempos de respuesta mayores a 3 segundos.
* La extracción de datos desde las máquinas debe realizarse en intervalos de tiempo definidos (por ejemplo, cada 5 minutos) sin pérdida de datos.

**Criterios de Seguridad**

* Todas las conexiones deben realizarse a través de HTTPS.
* Los tokens de autenticación deben ser válidos y gestionarse correctamente (expiración, renovación).
* Los datos sensibles deben estar cifrados en la base de datos.

**5. Roles y Responsabilidades**

* **Equipo de Desarrollo**: Responsable de la corrección de defectos y la implementación de mejoras según los resultados de las pruebas.
* **Equipo de QA (Quality Assurance)**: Ejecutará las pruebas, documentará los defectos y verificará la corrección de los errores.
* **Product Owner**: Asegurará que todas las funcionalidades críticas hayan sido correctamente probadas y aprobadas antes de la implementación final.

# Planificación

**Fechas Importantes**

* **Inicio de pruebas**: 8 de octubre
* **Finalización de pruebas**: 15 de noviembre
* **Pruebas de aceptación del usuario**: 20 de noviembre

**Recursos Necesarios**

* Dispositivos móviles (Android e iOS) para probar la aplicación en diferentes plataformas.
* Servidores para el backend y la base de datos.
* Herramientas de monitoreo y análisis de rendimiento (por ejemplo, **Postman** – RestClient para pruebas de API.

# Entregables

* **Informe de Pruebas**: Un documento detallado que resuma los resultados de las pruebas, incluyendo defectos encontrados, tiempo de ejecución y recomendaciones para la implementación final.
* **Lista de Defectos**: Un listado de todos los defectos encontrados, clasificados por severidad y prioridad.
* **Plan de corrección**: Un cronograma de las correcciones que el equipo de desarrollo implementará para resolver los defectos encontrados.

# Riesgos

**Posibles Riesgos**

* Retrasos en la ejecución de las pruebas debido a problemas de integración con las máquinas.
* Fallos en la extracción de datos debido a inconsistencias en el protocolo de comunicación con las máquinas.

**Plan de Mitigación**

* Hay que asegurar que se cuenta con la documentación técnica completa de las máquinas para garantizar que la integración se realice correctamente.
* Realizar pruebas preliminares de extracción de datos para detectar posibles inconsistencias antes de iniciar las pruebas completas.