

Plan de Pruebas

## Proyecto: Sistema de Información para la Gestión de Devolución de Uniformes Johan Uniforms

## Versión: 1.1

Historia de revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Autor(es) | Descripción | Fecha |
| 1.0 | Valentina Vasquez | Planificación del sistema y desarrollo del módulo de registro de usuarios. | Abril/08/2024 |
| 1.1 | Leidy Vallejo | Implementación de APIs externas e internas como (validación de correos y recuperar). | Mayo/17/2024 |
| 1.2 | Laura Diaz | Desarrollo del módulo CRUD para gestión de productos y devoluciones | Mayo/29/2024 |
| 1.3 | Luisa Castillo | Integración de roles (cliente y administrador) con accesos diferenciados. | Junio/17/2024 |
| 1.4 | Equipo Completo | Integración de funcionalidades: inicio de sesión, recuperación de contraseña, carrito de compras, facturación y generación de PDF | Junio/28/2024 |

## Índice

1. ​Introducción
   1. ​Alcance
   2. ​Referencias
   3. ​Glosario
2. ​Contexto de las Pruebas
   1. ​Proyecto / Subprocesos de Prueba
   2. ​Elementos de Prueba
   3. ​Alcance de la Prueba
   4. ​Suposiciones y Restricciones
   5. ​Partes Interesadas
3. ​Comunicación de las Pruebas
4. ​Registro de Riesgos
5. ​Estrategia de Prueba
   1. ​Subprocesos de prueba
   2. ​Entregables de Prueba
   3. ​Técnicas de diseño de Prueba
   4. ​Criterio de Finalización y Prueba
   5. ​Métricas
   6. ​Requisitos del entorno de Pruebas
      1. ​Ambiente de pruebas
      2. ​Herramientas de Pruebas
   7. ​Re-testing y regresión de las Pruebas
   8. ​Criterios de Suspensión y Reanudación
      1. ​Criterios de suspensión
      2. ​Criterio de reanudación
   9. ​Desviaciones de la Estrategia de Prueba Organizacional
6. ​Actividades y Estimados de Prueba
7. ​Personal
   1. ​Roles, Actividades y Responsabilidades
   2. ​Necesidades de Contratación
   3. ​Necesidades de Entrenamiento 8 Cronograma

# Introducción

### Alcance

El objetivo de este documento es brindar la información acerca de los elementos técnicos y conceptuales que sean requeridos para planificar y desarrollar las actividades del proceso de pruebas del sistema de información Johan Uniforms

### Referencias

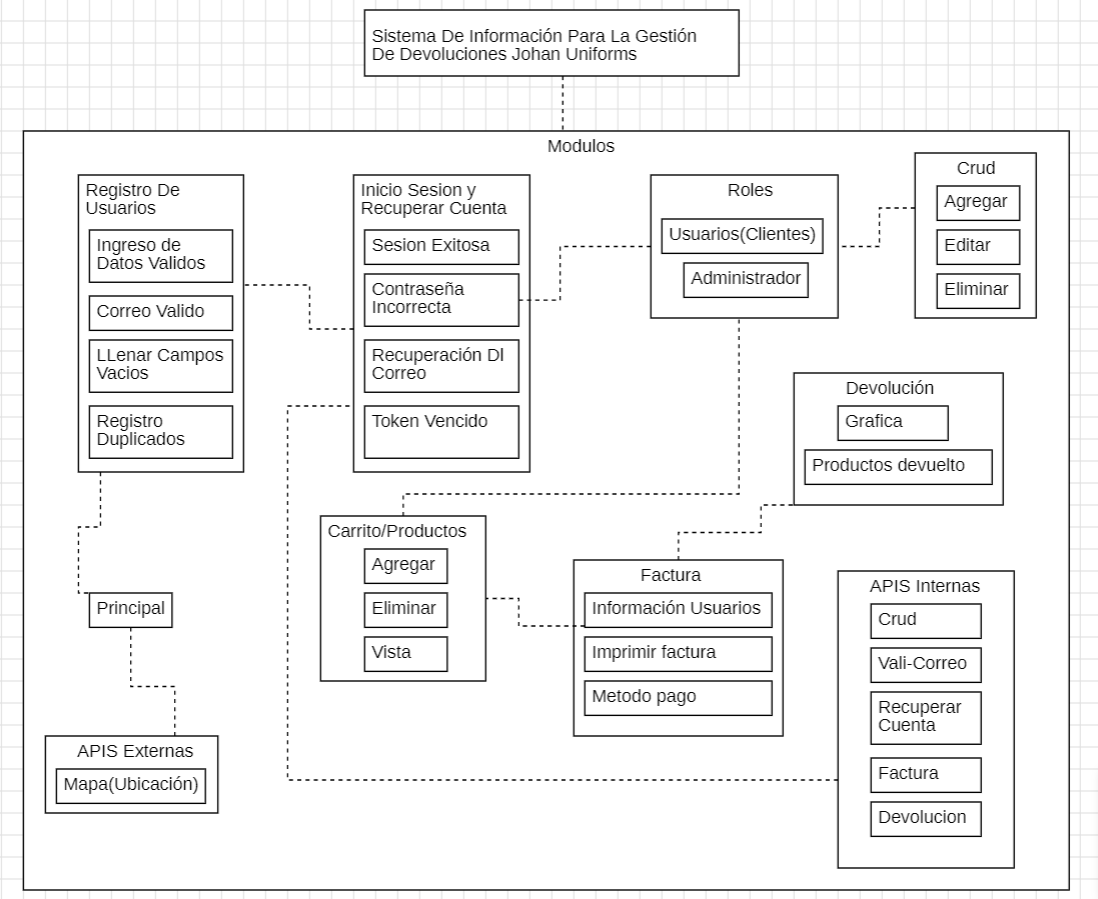
* Requisitos funcionales y no funcionales del proyecto
* Arquitectura y diseño del sistema
* ISO 29119

### Glosario

|  |  |
| --- | --- |
| **Pruebas unitarias (Unit Testing)**  **Imagen que contiene Icono  El contenido generado por IA puede ser incorrecto.** | Proceso de verificación en el que se evalúan individualmente las unidades más pequeñas del código (funciones, métodos o clases) para asegurar que funcionan correctamente de forma aislada, sin depender de otras partes del sistema. |
| **Pruebas de integración**  **Icono  El contenido generado por IA puede ser incorrecto.** | Tipo de prueba que valida la interacción entre múltiples unidades o módulos del sistema, comprobando que colaboran correctamente al integrarse y que los datos fluyen adecuadamente entre ellos. |
| **Caja negra (Black-box)**  **Imagen que contiene Interfaz de usuario gráfica  El contenido generado por IA puede ser incorrecto.** | Enfoque de prueba donde se evalúa la funcionalidad del software sin conocimiento interno de su código fuente o estructura. Se basa en las entradas y salidas esperadas, verificando el comportamiento externo del sistema |
| **Axios**  **Interfaz de usuario gráfica, Icono  El contenido generado por IA puede ser incorrecto.** | Librería basada en promesas para realizar solicitudes HTTP desde el navegador o Node.js. Facilita la comunicación con APIs, permite interceptar peticiones/respuestas y soporta operaciones asincrónicas como GET, POST, PUT y DELETE |
| **Fetch**  **Icono  El contenido generado por IA puede ser incorrecto.** | Interfaz nativa de JavaScript que permite realizar solicitudes HTTP de forma asincrónica. Utiliza promesas para manejar respuestas del servidor y es ampliamente utilizada en aplicaciones web modernas para consumir APIs |

# Contexto de las Pruebas

### Proyecto / Subprocesos de Prueba

c

### Elementos de Prueba

Se realizarán pruebas a los siguientes elementos:

* Módulo Registro de Usuarios.

|  |  |
| --- | --- |
| Objetivo | Pruebas a Realizar |
| Verificar que los usuarios puedan registrarse correctamente con datos válidos(reales). | Registro con datos validos.  Registro Fallido con correo invalido(validación con API).  Registros duplicados.  Campos vacío obligatorios. |

* Módulo Inicio sesión y Recuperar Cuenta.

|  |  |
| --- | --- |
| Objetivo | Pruebas a Realizar |
| Garantizar que los usuarios puedan iniciar sesión y recuperar su cuenta. | Inicio de sesión exitoso. Contraseña incorrecta. Recuperación de contraseña por correo. Token de recuperación vencido. |

* Módulo Crud de Productos.

|  |  |
| --- | --- |
| Objetivo | Pruebas a Realizar |
| Validar la gestión completa de los productos (crear, leer, actualizar, eliminar). | Creación de productos con datos completos. Edición de productos. Eliminación de productos con y sin restricciones. Visualización correcta en el catálogo. |

* Módulo Roles.

|  |  |
| --- | --- |
| Objetivo | Pruebas a Realizar |
| Confirmar que los accesos estén correctamente definidos para cliente y administrador. | El cliente solo puede ver y comprar productos. El administrador accede a funciones de gestión (CRUD, facturación). Acceso restringido a funciones no autorizadas. |

* Módulo de APIS.

|  |  |
| --- | --- |
| Objetivo | Pruebas a Realizar |
| Verificar el correcto funcionamiento de las integraciones externas. | Validación de correo electrónico real. Integración de métodos de pago (Nequi, efectivo). |

* Modulo Carrito de Compras y Factura.

|  |  |
| --- | --- |
| Objetivo | Pruebas a Realizar |
| Validar la funcionalidad del carrito y generación de factura en PDF. | Agregar y eliminar productos del carrito. Generar PDF con datos correctos.  Grafica de devolución. |

### Alcance de la Prueba

Las pruebas de software se clasifican según diversos criterios, entre ellos, el momento en que se aplican. En este contexto, se implementaron tanto pruebas estáticas como pruebas dinámicas, cada una con su respectivo enfoque y propósito.

En primer lugar, se llevaron a cabo pruebas estáticas, aplicadas durante las fases iniciales del desarrollo. Estas pruebas se ejecutaron a nivel de:

* Pruebas unitarias (Unit Testing): donde se evaluó de forma individual cada módulo o componente del sistema, asegurando que su lógica interna funcionará correctamente. Para ello, se utilizó el enfoque de caja blanca (White-box), que permite examinar la estructura y el flujo del código fuente.
* Pruebas de integración: enfocadas en verificar la correcta interacción entre múltiples módulos del sistema. También se aplicó el enfoque de caja blanca, asegurando que la integración entre componentes se desarrollara conforme a lo esperado.

Posteriormente, se realizaron pruebas dinámicas, las cuales implican la ejecución del software para analizar su comportamiento en tiempo real, aunque aún no esté en producción. En este caso, se empleó el enfoque de caja negra (Black-box), en el cual se validan las entradas y salidas del sistema sin tener en cuenta su lógica interna, enfocándose exclusivamente en la funcionalidad esperada desde la perspectiva del usuario.

### Suposiciones y Restricciones

* Suposición: Por medio de pruebas realizadas en JIRA, se tuvo en cuenta el

funcionamiento efectivo del sistema en la cual se pudo comprobar la efectividad

de ella por medio del realizador de pruebas.

* Restricciones: se realizaron pruebas por cada colaborador por medio de meet

cada 2 horas en hora de conectarse todos los colaboradores es de 12:30 pm

que se lleva a cabo como xsz. Se realizará durante el día completo por 3 días

consecutivos para la valoración de las pruebas.

* 1. Partes Interesadas

|  |  |
| --- | --- |
| Cliente | Aprobación que se recibe por parte del cliente por el plan de pruebas de JIRA, y el funcionamiento continuado de aquella cumpliendo todas las funcionalidades en el cronograma establecido. |
| Empresa | El cumplimiento de las pruebas y que se esté cumpliendo el cronograma de aquella. |
| Programadores | Estar al tanto de las pruebas pero con más disponibilidad por fallos o inconvenientes con el proyecto que se está realizando las pruebas. |

# Comunicación de las Pruebas

Los detalles acerca de las responsabilidades de los clientes y miembros del equipo se detallan en este acápite.

Se describe el proceso para las comunicaciones internas, externas, el protocolo de comunicación y la resolución de conflictos.

A continuación se identifican los principales puntos de comunicación:

| **Punto de Comunicación** | **Propósito** | **Frecuencia** | **Medios** | **Responsable** | **Audiencia** |
| --- | --- | --- | --- | --- | --- |
| Reunión de inicio | Inicio del proyecto | Una vez | Encuentro personal | Líder de Proyecto/ QA Manager | Equipo |
| Informe de avances | Seguimiento de estado | Semanal | Reunión | Líder de Proyecto/ QA Lead | Equipo |
| Reportes de Estado | Reporte Documental | Semanal | Documento vía email | Líder de Proyecto/ QA Engineer | Equipo |
| Cierre de sprint | Evaluar resultados | Al finalizar cada sprint | Reunión | Líder de Proyecto/ QA Lead | Equipo |

# Registro de Riesgos

En la siguiente tabla se identifican los riesgos del proyecto, así como se determina la severidad de cada uno de los riesgos multiplicando el impacto por la probabilidad de ocurrencia.

El impacto y la probabilidad se determinan teniendo en cuenta una escala de 1 al 5, donde 5 es el más alto

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **⚠️Riesgos** | **🎲Probabilidad (1-5)** | **🎯Impacto (1-5)** | **🔥Severidad (Prob\*Impct)** | **🛡️Plan de Mitigación** |
| 1 | Retrasos en la implementación de las funcionalidades. | 2 | 5 | 🟢10 | Evaluar el avance del desarrollo de las funcionalidades y re-planificar acorde al avance de ser necesario. |
| 2 | Campos que no validan correctamente (al momento de insertar un teléfono que acepte letras) | 4 | 2 | 🟡8 | Identificar el segmento donde se presenta el incidente. |
| 3 | Módulos que tardan más de lo deseado en cargar, pero no se caen | 2 | 4 | 🟡9 | Verificar el rendimiento del sistema |
| 4 | Acepta devoluciones de productos equivocados por el usuario | 1 | 5 | 🟢10 | Evaluar la funcionalidad del módulo y de igual manera ejecutar pruebas necesarias |

# Estrategia de Prueba

Se enfoca en la **verificación funcional** del sistema de manera modular y progresiva, para asegurar una experiencia estable y segura para el usuario.

**🔍Tipo de prueba**

Pruebas **funcionales** centradas en entradas/salidas esperadas.  
 No se considera la lógica interna del código (caja negra).

**🧩Cobertura de módulo**

Cada módulo será evaluado **de forma independiente**:

|  |  |
| --- | --- |
| **Modulo** | **Elementos evaluados** |
| Registro | Validación de campos, duplicados |
| Inicio sesión | Errores en credenciales, permiso |
| Crud | Operaciones básicas, restricciones |
| Roles | Accesos y visibilidad |
| Carrito | Agregado/elminado |
| API | Pruebas cons postman |
| Facturación | validación totales. |

**✅Casos positivos y negativos**

Se evaluarán escenarios como:

* Campos vacíos o incorrectos
* Errores en credenciales
* Registros duplicados
* Accesos no permitidos

**⚙️Automatización parcial**

Herramientas como **Postman** serán usadas para validar endpoints de la API.

**🔗Validación de integraciones**

Se comprobará el correcto funcionamiento de integraciones con servicios externos (verificación de correos reales, medios de pago).

✅ Verificación de correos reales

💳 Medios de pago (Nequi, PayPal, Daviplata, etc.)

### Subprocesos de prueba

Las pruebas para el Sistema de información para la gestión de devoluciones Johan Uniforms incluirán los siguientes subprocesos de prueba:

|  |  |
| --- | --- |
| **Sbproceso** | **Descripción** |
| 🧱Pruebas de componentes | Verificación individual de cada módulo o funcionalidad del sistema (ej. login, carrito). |
| 🔗 Pruebas de integración | Evaluación de la comunicación entre módulos (ej. registro + API + base de datos). |
| 🧪Pruebas de sistema | Pruebas del sistema completo en un entorno simulado, como lo usaría un usuario real. |

### Entregables de Prueba

Cada subproceso de prueba generará los siguientes documentos clave, que permiten asegurar trazabilidad y control en la validación del sistema:

|  |  |
| --- | --- |
| 📄 **Entregable** | 📝 **Descripción** |
| 📘 Plan de Pruebas del Subproceso | Define el enfoque, criterios de entrada/salida, responsables y cronograma. |
| 🧪 Especificación de Casos de Prueba | Lista detallada de los casos que se ejecutarán, con condiciones, entradas y salidas esperadas. |
| 📊 Informe de Estado de las Pruebas | Resumen del progreso, resultados obtenidos, bloqueos y cobertura de pruebas. |
| ✅ Informe de Finalización del Subproceso | Documento que confirma la conclusión del subproceso de pruebas, análisis de resultados y lecciones aprendidas. |

### Técnicas de diseño de Prueba

En este acápite se identifican las técnicas que se utilizarán para el diseño de las pruebas.

* Pruebas de Casos de Uso
* Particiones de Equivalencia y Análisis de Valores Límites.
* Pruebas Exploratorias

### Criterio de Finalización y Prueba

Las pruebas deben alcanzar una cobertura de requisitos del 80% y todos los procedimientos de pruebas deben ejecutarse sin fallas de gravedad 1 (alta).

* Se ejecutaron el 100% de los casos de prueba planificados.
* El porcentaje de pruebas exitosas es mayor al 90%.
* Todos los errores críticos y altos fueron corregidos y validados.
* Se generó el informe final de pruebas.

### Métricas

Las siguientes métricas se recogerán durante el transcurso de la ejecución de las pruebas:

* Número de casos de prueba ejecutados.
* Número de incidentes por categoría.
* Número de casos de prueba re-ejecutados.
* Número de incidentes resueltos por categoría.
* Defectos encontrados post-release
* Cobertura de requerimientos

### Requisitos del entorno de Pruebas

* + 1. Ambiente de pruebas

|  |  |
| --- | --- |
| Navegadores | Chrome, Mozilla Firefox |
| Sistemas Operativos | Windows |

* + 1. Herramientas de Pruebas

|  |  |
| --- | --- |
| Herramienta | Función |
| Correo Electrónico Real | Pruebas de validación de registro y recuperación de cuenta. |
| Postman | Verificación manual de las APIs del sistema. |
| Capturas de Pantalla | Evidencia visual de resultados exitosos y fallidos de pruebas. |

### Re-testing y regresión de las Pruebas

pruebas de confirmación (*re-testing*) y pruebas de regresión durante el proceso de validación.

* **Primer Ciclo:** Enfocado en la validación inicial de funcionalidades nuevas y ejecución de pruebas básicas.
* **Segundo Ciclo:** Se realizarán pruebas de re-testing para confirmar la corrección de defectos reportados en el primer ciclo, así como una regresión parcial de funcionalidades relacionadas.
* **Tercer Ciclo (Final):** Incluirá una prueba de regresión completa para asegurar que las correcciones y nuevas funcionalidades no hayan impactado negativamente otras partes del sistema.

### Criterios de Suspensión y Reanudación

* + 1. Criterios de suspensión
* El sistema no cumple con funcionalidades clave especificadas en los requisitos, como el registro de usuarios, el inicio de sesión, la gestión del carrito de compras, o la generación de facturas en PDF.
* Errores críticos en módulos que bloqueen o impidan el flujo normal de las pruebas y la interacción con otros módulos.
* Fallas en la integración de APIS, como la validación de correos electrónicos y los métodos de pago,

que afectan la funcionalidad de registro y compra.

* Inestabilidad en el entorno de pruebas, o problemas al acceder al sistema desde distintos navegadores o plataformas.
* Cambios no autorizados por parte del equipo de desarrollo que afecten las funcionalidades en el proyecto. Como modificaciones en la base de datos o en las APIS que interrumpen las pruebas programadas.

5.8.1 Criterio de reanudación

Llegar a un acuerdo entre las partes para reanudar las pruebas, además de condiciones mínimas como:

* Corrección de las fallas críticas que causaron la suspensión, asegurando que los módulos, funciones de forma correcta.
* Restablecimiento del entorno de pruebas, con los datos y configuraciones necesarias para asegurar que las pruebas se realicen bajo condiciones estables y consistentes.
* Validación de correos electrónicos y la integración con métodos de pago, lo cual permitirá retomar las pruebas de los módulos de registro y compras.
* Confirmación de que los cambios realizados por el equipo de desarrollo no afectaron negativamente las funcionalidades críticas y están debidamente documentados y validados.

Y por ultimo. **Aprobación formal** por parte del líder del proyecto para continuar con las pruebas, una vez que los errores hayan sido solucionados y el sistema esté listo para ser probado nuevamente.

### Desviaciones de la Estrategia de Prueba Organizacional

La Estrategia de Prueba Organizacional establece como objetivo una cobertura completa del 100% de los requisitos para asegurar la trazabilidad y la validación integral del sistema. No obstante, en esta etapa específica de pruebas, se ha decidido reducir la cobertura al 80%. Esta decisión se fundamenta en el análisis de riesgos, que indica una baja probabilidad e impacto de fallos críticos en los módulos evaluados. Además, se ha planificado llevar a cabo pruebas rigurosas y exhaustivas a nivel de componente, lo cual compensará la reducción en la cobertura global, garantizando así la calidad y la estabilidad del sistema en su conjunto.

# Actividades y Estimados de Prueba

Las pruebas se dividirán en las siguientes actividades principales:

1. Por medio de un diagrama de casos de uso en UML se hizo la simulación del sistema donde interactúan dos roles (Usuario, Administrador).

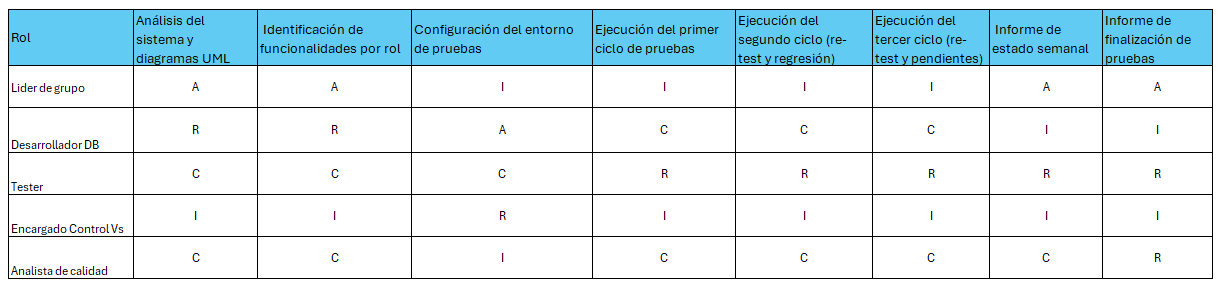
<https://github.com/LauraVDB567/Jhoan_uniformss/tree/main/Docs/2%20TRIMESTRE/2.2%20CASOS%20DE%20USO>

1. Es una serie de pasos que se llevan a cabo donde el administrador va a tener acceso a la gestión de los usuarios donde este podrá modificar o eliminar ya en el rol del usuario tendrá vista al registro e inicio sesión, recuperar, contraseña, carrito de compras, factura, devolución y solicitud.
2. En el backend se crea la api con sus respectivos métodos en este caso (Get, Post, Put, Delete) y el frontend se consumen ya sea con axios o fetch y por medio de Postman se hace las validaciones de estos métodos si cumplen con lo esperado.
3. **Primer Ciclo:** Se valida que el registro e inicio de sesión y otros módulos estén funcionando correctamente .
4. **Segundo ciclo :**En caso de presentarse algún tipo de error en el backend o frontend se volverían a realizar las respectivas pruebas y mirar en que está fallando.
5. **Tercer ciclo:** Según las nuevas modificaciones de los errores anteriores ya se probaría el sistema en general.
6. **Informe de reporte de estado de las pruebas semanal**: En el momento no se han encontrado ningún tipo de errores ya si se realizan más modificaciones podrán surgir conflictos o errores .
7. **Informe de finalización de las pruebas:** Por medio de un cronograma de actividades se da a entender que los resultados ya están hechos solo faltaría las nuevas modificaciones a realizar

# Personal

### Roles, Actividades y Responsabilidades

La matriz RACI (Responsible-Accountable-Consulted-Informed) a continuación ilustra qué rol está involucrado en qué actividad (es) y cuál es el nivel de participación.



<https://docs.google.com/spreadsheets/d/1l0SNc5g4MUr8OIBwc119mT_vXsq1v43r/edit?usp=sharing&ouid=105468914221315508197&rtpof=true&sd=true>

### Necesidades de Contratación

Para garantizar la adecuada verificación y validación de los productos de software generados, se prevé la necesidad de integrar al menos dos (2) Analistas de Calidad (Quality Assurance, QA), cuya función principal será la ejecución sistemática de pruebas funcionales y no funcionales, conforme a los criterios de aceptación previamente definidos. Adicionalmente, se contempla la incorporación de un (1) Ingeniero de Automatización de Pruebas, encargado del diseño e implementación de marcos de automatización que permitan la ejecución repetitiva y eficiente de casos de prueba, favoreciendo la detección temprana de defectos y la mejora continua del proceso de aseguramiento de calidad.

### Necesidades de Entrenamiento

Se capacitará al personal en la arquitectura, funcionalidades y flujos del software, detallando paso a paso los módulos a desarrollar conforme a los requerimientos técnicos. Esta capacitación incluirá la revisión de los criterios de aceptación y casos de prueba, con el objetivo de validar los requisitos previamente definidos y asegurar su correcta implementación en el entorno de desarrollo.

# Cronograma

En este cronograma se puede evidenciar las actividades ya realizadas con anterioridad y con su respectiva fecha de inicio y finalización

Calendario

El contenido generado por IA puede ser incorrecto.