**Plan de pruebas de software**

***Heavy Metal***

***Fecha: 15/05/2024***

**Tabla de contenido**

Historial de versiones 3

Información del proyecto 4

Aprobaciones 4

Resumen ejecutivo 4

Alcance de las pruebas 5

Elementos de pruebas 5

Funcionalidades a probar 5

Funcionalidades a no probar 5

Enfoque de pruebas (estrategia) 5

Criterios de aceptación o rechazo 6

Criterios de aceptación o rechazo 6

Criterios de suspensión 6

Criterios de reanudación 6

Entregables 6

Recursos 6

Requerimientos de entornos – Hardware 6

Requerimientos de entornos – Software 7

Herramientas de pruebas requeridas 7

Personal 7

Entrenamiento 7

Planificación y organización 8

Procedimientos para las pruebas 8

Matriz de responsabilidades 8

Cronograma 8

Glosario 8

# Historial de versiones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| **15/05/2024** | **1.0** | **Carlos Alberto Gaitan Moreno** | **Heavy Metal** | **Plan de Pruebas Heavy Metal** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Información del proyecto

|  |  |
| --- | --- |
| Empresa / Organización | SENA – CEET |
| Proyecto | Heavy Metal |
| Fecha de preparación | 15/05/2024 |
| Cliente | Heavy Metal |
| Patrocinador principal | SENA – CEET |
| Gerente / Líder de proyecto | Carlos Alberto Gaitán Moreno |
| Gerente / Líder de pruebas de software | Carlos Alberto Gaitan Moreno |

# Aprobaciones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nombre y Apellido** | **Cargo** | **Departamento u organización** | **Fecha** | **Firma** |
| **Daniel Mora** | **Instructor** | **Centro de Electricidad, Electronica y Telecomunicaciones** | **15/05/2024** |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Resumen ejecutivo

El propósito de este documento es proporcionar un resumen del Plan de Pruebas para la API desarrollada con Django, que se consume en Angular y utiliza MySQL como motor de base de datos. Este es un plan detallado que identifica el alcance del esfuerzo de pruebas en relación con el plan de proyecto de software.

El objetivo principal de este plan de pruebas es garantizar que la API cumpla con los requisitos funcionales y no funcionales especificados, y que ofrezca un rendimiento óptimo cuando se consume en Angular. Las pruebas se realizarán en varios niveles, incluyendo pruebas unitarias, pruebas de integración y pruebas de sistema.

Este plan de pruebas tiene un alcance definido que incluye la validación de todas las funcionalidades de la API, así como la verificación de la correcta interacción con la base de datos MySQL. Sin embargo, debido a restricciones de recursos y presupuesto, algunas áreas pueden tener un alcance de prueba limitado.

Es importante destacar que este plan de pruebas está sujeto a cambios basados en los hallazgos durante el ciclo de pruebas y la retroalimentación de los stakeholders. El equipo de pruebas trabajará en estrecha colaboración con los desarrolladores y los stakeholders para garantizar que se mantenga la calidad del software y se cumplan los objetivos del proyecto.

# Alcance de las pruebas

## Elementos de pruebas

**Módulo de Gestión de Productos**: Este módulo se encarga de la administración de los productos de ferretería disponibles en la tienda virtual. Las pruebas se centrarán en la correcta gestión de los productos (agregar, editar, eliminar productos), así como en la correcta visualización de los productos en la tienda.

**Proceso de Compra a través del Carrito de Compras**: Este proceso implica la selección de productos, la adición de estos al carrito de compras y la finalización de la compra. Las pruebas verificarán que los usuarios puedan realizar estas acciones sin problemas y que los cálculos de precios sean correctos.

**Generación de Domicilios**: Este módulo se encarga de la creación y gestión de las direcciones de envío de los usuarios. Las pruebas se centrarán en la correcta creación, edición y eliminación de domicilios.

**Manejo de Gestión de Domicilios**: Este módulo se encarga de la asignación de domicilios a las compras realizadas. Las pruebas verificarán que los domicilios se asignen correctamente a las compras y que se puedan gestionar adecuadamente.

**Registro de Usuarios**: Este módulo se encarga de la creación de cuentas de usuario. Las pruebas se centrarán en la correcta creación de cuentas, así como en la validación de los datos de los usuarios.

**Visualización a Detalle por Producto**: Este módulo se encarga de mostrar los detalles de cada producto. Las pruebas verificarán que los detalles de los productos se muestren correctamente y que la información sea precisa.

**Login para el Ingreso**: Este módulo se encarga de la autenticación de los usuarios. Las pruebas se centrarán en la correcta autenticación de los usuarios y en la seguridad del proceso de login.

## Funcionalidades a probar

Búsqueda de Productos por Nombre**: Los usuarios deben poder buscar productos por nombre a través de la barra de búsqueda. Las pruebas verificarán que los resultados de la búsqueda sean precisos y relevantes.**

Filtro de Productos**: Los usuarios deben poder filtrar productos por rango de precios, categorías o marcas. Las pruebas verificarán que los filtros funcionen correctamente y que los resultados filtrados sean precisos.**

Registro de Usuarios**: Los usuarios deben poder registrarse en la plataforma. Las pruebas verificarán que el proceso de registro funcione correctamente y que los datos de los usuarios se almacenen de forma segura.**

Compra de Productos y Descuento del Stock**: Cuando un usuario compra un producto, el stock de ese producto debe disminuir en consecuencia. Las pruebas verificarán que el stock se actualice correctamente después de cada compra.**

## Funcionalidades a no probar

**Interfaz de Usuario (UI)**: Aunque la interfaz de usuario es una parte integral de la experiencia del usuario, este plan de pruebas se centrará en la funcionalidad de la API y no en la interfaz de usuario de la tienda virtual. La razón de esto es que la interfaz de usuario puede ser probada más eficientemente mediante pruebas de usuario y pruebas A/B. Sin embargo, al no probar la interfaz de usuario, corremos el riesgo de pasar por alto posibles problemas de usabilidad.

**Rendimiento bajo Carga Extrema**: No se realizarán pruebas de rendimiento bajo condiciones de carga extremadamente altas (por ejemplo, miles de usuarios simultáneos). La razón de esto es que se espera que la tienda virtual opere bajo un nivel de carga moderado y las pruebas de carga extrema requerirían recursos significativos. El riesgo aquí es que no sabremos cómo se comportará la tienda virtual bajo condiciones de carga extremadamente altas.

**Compatibilidad con Todos los Navegadores y Dispositivos**: No se probará la compatibilidad de la tienda virtual con todos los posibles navegadores y dispositivos. La razón de esto es que se espera que la mayoría de los usuarios utilicen los navegadores y dispositivos más comunes y probar todos los posibles navegadores y dispositivos requeriría una cantidad significativa de tiempo y recursos. El riesgo aquí es que algunos usuarios pueden experimentar problemas si utilizan un navegador o dispositivo no probado.

## Enfoque de pruebas (estrategia)

La estrategia de pruebas para el proyecto “Heavy Metal” se centrará en tres tipos de pruebas principales: pruebas funcionales, pruebas de seguridad y pruebas de usabilidad. Estas pruebas se han seleccionado para maximizar la cobertura de las funcionalidades clave del proyecto con un mínimo de recursos.

**Pruebas Funcionales**: Estas pruebas se centrarán en verificar que todas las funcionalidades de la tienda virtual funcionen como se espera. Esto incluirá pruebas de la funcionalidad de búsqueda de productos, filtrado de productos, registro de usuarios, y el proceso de compra y descuento del stock de productos.

**Pruebas de Seguridad**: Estas pruebas se centrarán en asegurar que la tienda virtual sea segura para los usuarios. Esto incluirá pruebas de la seguridad del proceso de registro y login, así como pruebas de la seguridad de los datos de los usuarios.

**Pruebas de Usabilidad**: Estas pruebas se centrarán en asegurar que la tienda virtual sea fácil de usar para los usuarios. Esto incluirá pruebas de la usabilidad de la interfaz de usuario, así como pruebas de la facilidad de uso de las funcionalidades clave de la tienda.

Para cada tipo de prueba, se utilizarán herramientas y técnicas que no requieran de un alto nivel de conocimientos técnicos o de licencias costosas. Por ejemplo, las pruebas funcionales y de usabilidad se pueden realizar manualmente por el equipo de pruebas, mientras que las pruebas de seguridad se pueden realizar utilizando herramientas de seguridad gratuitas disponibles en línea.

Es importante destacar que esta estrategia de pruebas es flexible y puede adaptarse según las necesidades del proyecto a medida que avanza.

# Criterios de aceptación o rechazo

## Criterios de aceptación o rechazo

Completar 100% de Pruebas Unitarias: Todas las pruebas unitarias definidas en el plan de pruebas deben ser ejecutadas y pasar exitosamente.

Porcentaje de Casos Exitosos: Un cierto porcentaje (por ejemplo, 95%) de todos los casos de prueba deben pasar exitosamente. Este porcentaje puede variar dependiendo de la criticidad de las funcionalidades y los riesgos asociados.

Porcentaje de Defectos Corregidos: Un cierto porcentaje (por ejemplo, 90%) de todos los defectos encontrados durante las pruebas deben ser corregidos y verificados.

Pruebas de Seguridad Exitosas: Todas las pruebas de seguridad definidas en el plan de pruebas deben ser ejecutadas y pasar exitosamente.

Pruebas de Usabilidad Exitosas: Todas las pruebas de usabilidad definidas en el plan de pruebas deben ser ejecutadas y pasar exitosamente.

## Criterios de suspensión

Existencia de Defectos Críticos: Si durante la ejecución de los casos de pruebas se descubre un defecto crítico que impide la funcionalidad básica del software, se detendrán las pruebas hasta que dicho defecto sea corregido y verificado.

Porcentaje de Casos Fallidos: Si un cierto porcentaje (por ejemplo, 30%) de los casos de prueba fallan, se detendrán las pruebas para investigar y corregir los problemas subyacentes antes de continuar.

Problemas de Seguridad Graves: Si se descubre un problema de seguridad grave durante las pruebas, se detendrán las pruebas hasta que el problema de seguridad sea resuelto.

Estas condiciones aseguran que los problemas graves sean abordados de inmediato y que los recursos de pruebas no se desperdicien en un software que tiene problemas conocidos graves.

## Criterios de reanudación

Corrección de Defectos Críticos: Las pruebas se reanudarán una vez que los defectos críticos que causaron la suspensión de las pruebas hayan sido corregidos y verificados.

Reducción del Porcentaje de Casos Fallidos: Las pruebas se reanudarán una vez que el porcentaje de casos de prueba fallidos se haya reducido a un nivel aceptable (por ejemplo, menos del 10%).

Resolución de Problemas de Seguridad Graves: Las pruebas se reanudarán una vez que los problemas de seguridad graves hayan sido resueltos.

Estos criterios aseguran que los problemas graves sean resueltos antes de continuar con las pruebas.

# Entregables

**Documento de Plan de Pruebas**: Este documento detalla la estrategia de pruebas, incluyendo los tipos de pruebas a realizar, los criterios de finalización y reanudación de las pruebas, y las funcionalidades a probar y no probar.

**Casos de Pruebas**: Este es un conjunto de condiciones o variables bajo las cuales un tester determinará si una aplicación, software o uno de sus features funciona como se esperaba.

**Logs de Errores**: Estos son registros de los errores encontrados durante las pruebas.

**Reportes de Incidencias**: Estos informes detallan los problemas encontrados durante las pruebas, incluyendo información sobre cómo se encontró el problema, cómo afecta al software y las recomendaciones para corregirlo.

**Evidencias de Pruebas**: Estas pueden incluir capturas de pantalla, videos, o cualquier otro tipo de evidencia que demuestre el resultado de las pruebas.

**Reportes Emitidos por Herramientas de Pruebas**: Si se utilizan herramientas de pruebas gratuitas, estos reportes proporcionarán información adicional sobre los resultados de las pruebas.

# Recursos

## Requerimientos de entornos – Hardware

Computador Portátil**: Un computador portátil con un procesador Intel i5 y 16 GB de RAM debería ser suficiente para ejecutar las pruebas. Este equipo se utilizará para ejecutar las pruebas y para acceder a las herramientas de pruebas.**

Sistema Operativo**: Windows 10 será el sistema operativo utilizado para ejecutar las pruebas y las herramientas de pruebas.**

Conectividad a la Red**: Se necesitará una conexión a Internet estable para acceder a la API y a la tienda virtual, así como para descargar y utilizar las herramientas de pruebas gratuitas disponibles en la web.**

Servidor de Aplicación**: Para la API desarrollada con Django, se puede utilizar un servidor de aplicación local como el servidor de desarrollo incorporado en Django para las pruebas.**

Base de Datos**: Para la base de datos MySQL, se puede utilizar una instancia local de MySQL para las pruebas. Asegúrate de tener suficiente espacio en disco para la base de datos.**

Herramientas de Pruebas Gratuitas**: Se utilizarán herramientas de pruebas gratuitas disponibles en la web. Algunas opciones podrían incluir Postman para pruebas de API, Selenium para pruebas de interfaz de usuario, y OWASP ZAP para pruebas de seguridad.**

## Requerimientos de entornos – Software

Sistema operativo: Windows 10

Django

MySQL

Angular

VS Code

**Postman**

**Selenium**

**OWASP ZAP**: OWASP ZAP

## Herramientas de pruebas requeridas

Postman: Postman es una herramienta de software gratuita que se utilizará para realizar pruebas de API. Permite enviar solicitudes HTTP a la API y ver las respuestas, lo que facilita la prueba de las funcionalidades de la API.

Selenium: Selenium es una herramienta de automatización de pruebas que se utilizará para realizar pruebas en la interfaz de usuario de la tienda virtual. Permite automatizar las acciones del usuario en el navegador y verificar que la interfaz de usuario funcione como se espera.

OWASP ZAP: OWASP ZAP es una herramienta de software gratuita que se utilizará para realizar pruebas de seguridad. Permite identificar posibles vulnerabilidades de seguridad en la tienda virtual y la API.

Metodología de Pruebas Ágiles: Se utilizará una metodología de pruebas ágiles para las pruebas. Esto significa que las pruebas se realizarán de manera iterativa y continua a lo largo del ciclo de desarrollo del software. Esta metodología permite identificar y corregir los problemas más rápidamente.

Técnicas de Pruebas Exploratorias: Las pruebas exploratorias son una técnica de pruebas que se utilizará para explorar la tienda virtual y la API y buscar posibles problemas. Esta técnica es útil para identificar problemas que no se habían considerado durante la planificación de las pruebas.

## Personal

## Carlos Alberto Gaitan Moreno Lider de Pruebas

## Daniel Loaiza Tester

## Entrenamiento

Necesidades de entrenamiento en el sistema o aplicación, así como en las herramientas de prueba a utilizar.

# Planificación y organización

## Procedimientos para las pruebas

Especifica los procedimientos o metodología de pruebas a emplear durante la ejecución del plan de pruebas de software.

## Cronograma

Debe estar basado en estimaciones de actividades realizadas por el equipo de prueba. En él se identifican los hitos relevantes en las pruebas de software, se establecen las dependencias (actividades predecesoras) y demás aspectos componentes de un cronograma.

# Glosario

Definiciones de términos usados en la documentación, y general sobre el área de pruebas.