**Plan de Pruebas**

**“***Sistema de administración y venta de productos para pequeñas empresas***”**

***[SAVPPE]***

***Fecha:[dd/mm/aaa]***

Histórico de Revisiones

| Versión | Fecha | Descripción/cambio | Autor |
| --- | --- | --- | --- |
| 001 |  | Alcance de las pruebas, descripción y resumen de las pruebas |  |
| 002 |  | Entorno y configuración de las pruebas, calendarización de las actividades de pruebas y resumen de riesgos |  |
| 003 |  | Definición de artefactos, condiciones de aceptación para cierre del proceso de pruebas y glosario |  |

Información del Proyecto

| Organización | Duoc UC. Escuela de Informática y Telecomunicaciones |
| --- | --- |
| Sección | PTY4614-004D |
| Proyecto (Nombre) | Savppe |
| Fecha de Inicio |  |
| Fecha de Término |  |
| Patrocinador principal |  |
| Docente | Julio Tapia |

Integrantes

| Rut | Nombre | Correo |
| --- | --- | --- |
| **21.248.045-2** | **Carolina Ahumada** | **caro.ahumada@duocuc.cl** |
| **21.037.262-8** | **Paolo Cozano** | **pa.cozano@duocuc.cl** |
| **20.558.995-3** | **Nallely Guerrero** | **nal.guerrero@duocuc.cl** |
| **17.924.335-0** | **Yohans Santa Maria** | **y.santamaria@duocuc.cl** |

| Alcance de las pruebas  *Definición de requisitos de S.W., módulos de Software a probar, Requisitos ambiente de pruebas y Documentación Referenciada, etc.* |
| --- |
| El objetivo del plan de pruebas es validar el correcto funcionamiento del sistema integral para la administración y venta de productos alimenticios cercanos a su fecha de vencimiento, incluyendo la página web y la aplicación de escritorio.  Requisitos de software a probar:   * Página web para ventas: visualización de productos, compra, y descuentos. * Aplicación de escritorio para administración: control de inventario, boletas y alertas. * Base de datos: integridad y sincronización de datos. * Integración con APIs: autenticación, dólar, comunas, mapa   Requisitos de ambiente de pruebas:   * Software: Visual Studio, Visual Studio Code, SQLite, navegadores (Chrome, Firefox), Jmeter y Selenium. * Hardware: PC con 8GB RAM, procesador i5 o superior, y conexión a Internet de alta velocidad.   Documentación referenciada:   * Requisitos del sistema. * Historias de usuario. * Especificaciones técnicas. |

| Descripción  *Descripción general del sistema y características clave* |
| --- |
| El sistema integral consta de dos componentes principales:  Página web para clientes desarrollada en el editor de código Visual Studio Code con el lenguaje de programación Python, desplegada en un entorno accesible, que permite a los usuarios explorar productos, aprovechar descuentos y realizar compras.  Aplicación de escritorio para administrador (Dueño de la tienda) construida con Visual Studio utilizando lenguaje de programación C#, que ofrece funcionalidades avanzadas para administrar el inventario, gestionar órdenes, y recibir alertas automáticas para productos próximos a vencer.  Para la base de datos se optó por SQLite, una solución que facilita el manejo de datos estructurados, ideal para aplicaciones de menor escala como esta o que requieren  una configuración rápida y sencilla.  Además, se empleó GitHub para el control de versiones, permitiendo una colaboración fluida entre los desarrolladores y se integraron diversas APIs para enriquecer el sistema con servicios externos. |

| Resumen de las pruebas | | |
| --- | --- | --- |
| Módulos del sistema a probar | Visualización de productos  Gestión de órdenes (carrito de compras) Aplicación de descuentos  Historial de compras  Métodos de pago  Autenticación cliente  Generación y descarga de boleta  Envío de productos  Control de inventario  Alertas de vencimiento y descuentos  Recepción de órdenes de compra  Visualización de boletas  Tutoriales de funcionamiento del sistema Autenticación administrador | |
| Objetivos de las pruebas | Identificar defectos y errores en el sistema.  Asegurar el cumplimiento de los requisitos funcionales.  Validar la experiencia de usuario en ambas plataformas.  Verificar la seguridad e integridad de los datos almacenados. | |
| Tipos de pruebas a realizar | Pruebas funcionales  Pruebas de carga y estrés | |
| Técnicas de pruebas a utilizar | Pruebas automatizadas para escenarios repetitivos con Selenium y Jmeter | |
| Roles y responsabilidades | Rol | Responsabilidades |
| Analista de sistemas / Diseñador | Diseñar y ejecutar historias de usuario. |
| Ingeniero en software / Tester | Resolver defectos reportados. |
| Product Owner | Asegurar el flujo adecuado del proceso de pruebas. |
| Scrum Master | Validar la aceptación del producto final. |

| Entorno y configuración de las pruebas  *Definir los requisitos de software y hardware necesarios para ejecutar las pruebas.* |
| --- |
| Hardware:  Computador con procesador i5, 8GB de RAM y disco SSD.  Red local para sincronización.  Software:  Visual Studio, Visual Studio Code.  SQLite para la base de datos.  Navegadores web actualizados.  Selenium y Jmeter para pruebas de historias de usuarios. |

| Resumen de riesgos  *Listado de riesgos relacionados al proceso de pruebas de S.W. Indicar riesgo, magnitud o impacto de este riesgo por etapa en el proceso. Magnitud: Alto, Significativo , Moderado, Inferior y Baja. Probabilidad de ocurrencia. Plan de mitigación y plan de contingencia.* |
| --- |

| Riesgo | Magnitud | Ocurrencia | Plan de Mitigación | Plan de Contingencia |
| --- | --- | --- | --- | --- |
| Errores en la sincronización de datos | Alto | Moderada | Revisar integraciones SQLite y depurar | Implementar almacenamiento temporal local |
| Caída del sistema durante pruebas de carga | Significativa | Baja | Facilitar consultas y mejorar configuración | Escalar hardware temporalmente |
| Errores en la aplicación de descuentos | Moderada | Moderada | Verificar algoritmos de descuento | Alertar al usuario y desactivar descuentos |
| API inaccesible | Significativa | Moderada | Configurar API’s alternativas | Usar valores almacenados en caché |

| Definición de artefactos  *Listar y describir los artefactos que serán administrados y entregados durante este proceso de prueba.* | |
| --- | --- |
| Artefacto | Descripción |
| Historias de usuario | Documento con los escenarios, criterios de aceptación y pasos a validar. |
| Script de pruebas | Código que automatiza las pruebas funcionales de la historia de usuario. |
| Gráficas de análisis de rendimiento | Visualización de datos de carga, porcentaje de errores y rendimiento del servidor. |
| Reporte HTML | Informe detallado de los resultados de ejecución, mostrando pasos aprobados o fallidos. |

| Condiciones de aceptación para cierre del proceso de pruebas  *Condiciones que se deben cumplir para dar termino al proceso de pruebas y margen de tolerancia de aceptación de defectos.* |
| --- |
| Todas las funcionalidades críticas deben pasar las pruebas sin defectos de alta magnitud.  Defectos menores deben tener un margen de tolerancia.  El sistema debe cumplir con los requisitos funcionales definidos.  Deben completarse al menos el 95% de los casos de prueba planificados. |

| Glosario |
| --- |
| **API**: Interfaz de programación de aplicaciones que permite comunicación entre diferentes sistemas.  **SQLite**: Sistema de base de datos embebido ligero.  **Defecto**: Error encontrado durante las pruebas. |