**Plan de pruebas de software**

**[Italia Pizza]**

**Fecha: [23/05/2024]**

**Equipo 1:**

**[Sujey Alarcón Hernández,**

**Eduardo Carrera Colorado,**

**Aneth Michelle Tamariz Moreno]**

**Tabla de contenido**

Historial del versiones del documento 3

PARTE I. PLAN DE PRUEBA 4

1. Introducción 4

1.1 Objetivos del plan de pruebas 5

1.2 Documentos relacionados 5

1.3 Glosario 7

2. Alcance de las pruebas 8

2.1 Elementos de pruebas 8

2.2 Funcionalidades a probar 8

2.3 Funcionalidades a no probar 9

3. Criterios 9

3.2 Criterios de inicio 9

3.3 Bases de datos para las pruebas 10

3.4 Criterios de aceptación o rechazo 10

4. Recursos 10

4.1 Requerimientos de entorno – Hardware 10

4.2 Requerimientos de entorno – Software 11

4.3  Personal 11

5. Planificación y organización 11

5.1 Procedimientos para las pruebas 11

5.2 Casos de pruebas 12

5.3 Cronograma 12

5.4 Dependencias y riesgos 12

PARTE II. REPORTE DE PRUEBA 14

6. Actividades desarrolladas 14

6.1 Cambios 14

6.2 Cronograma 14

7. Resultados 15

7.1 Principales incidencias 15

7.2 Cálculo de métricas 15

7.3. Análisis de la métricas 15

8. Conclusiones 17

8.1 Principales conclusiones 17

8.2 Recomendaciones para la mejora del proceso 17

9. Referencias 18

ANEXO A. Casos de prueba 19

ANEXO B. Script para pruebas. 19

# Historial del versiones del documento

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Versión** | **Fecha** | **Apartado** | **Cambios** | **Elaborado** | **Revisado** | |
| 1.0 | 23/05/2024 | Parte I | Definir secciones del plan, estructurar documento | 23/05/2024 |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# PARTE I. PLAN DE PRUEBA

# 1. Introducción

Italia Pizza es una iniciativa encabezada por un equipo de desarrolladores comprometidos con la modernización de la gestión empresarial en el sector de la industria gastronómica. El equipo, conformado por Sujey Alarcón Hernández, Eduardo Carrera Colorado y Aneth Michelle Tamariz Moreno tienen la tarea de crear un sistema de gestión para afrontar los desafíos operativos que enfrenta Italia Pizza, una empresa dedicada al servicio de comida. Italia Pizza opera en un entorno donde la gestión manual de registros ha demostrado ser ineficiente y propenso a errores. La carencia de un sistema de información adecuado ha generado problemas significativos, como un control deficiente sobre los productos, pedidos y usuarios, así como una pérdida considerable de tiempo en la reconciliación y seguimiento de pedidos. Este proyecto se enfoca en el ámbito específico del negocio de restaurantes de comida rápida, con especial atención en la gestión de pedidos, inventario, usuarios y finanzas. Para abordar estas necesidades, se emplearán tecnologías clave como el Framework .NET, Entity Framework, WPF (Windows Presentation Foundation) y SQL Server, garantizando así un desarrollo eficiente y escalable del sistema.

|  |  |  |  |
| --- | --- | --- | --- |
| Proyecto(s) | | Tipo de proyecto | |
| Italia Pizza | | Aplicación de escritorio | |
| Documentos relacionados | | | |
| Repositorio de documentación | | <https://github.com/fairyofshampoo/Italia-Pizza-Documents.git> | |
| **Equipo del proyecto** | | | |
| Jefe del equipo | Eduardo Carrera Colorado | Arquitecto del producto | Aneth Michelle Tamariz Moreno, Sujey Alarcón Hernández, Eduardo Carrera Colorado |

## 1.1 Objetivos del plan de pruebas

El objetivo general es garantizar que el sistema de gestión Italia Pizza funcione correctamente y cumpla con todos los requisitos especificados, proporcionando una experiencia de usuario fluida y eficiente para la gestión de productos, pedidos, usuarios y finanzas.

**Objetivos específicos:**

* Asegurar que todas las operaciones de registro, edición y eliminación de productos, pedidos y usuarios se realicen correctamente, manteniendo la integridad y consistencia de la base de datos.
* Probar que el sistema pueda manejar eficientemente un gran volumen de datos y múltiples operaciones simultáneas sin degradar el rendimiento.
* Comprobar que el sistema pueda generar reportes precisos en formato PDF, incluyendo reportes de inventario y cortes de caja de manera correcta con los datos almacenados en el sistema.
* Verificar que las funciones de búsqueda y filtrado operen correctamente, permitiendo a los usuarios localizar fácilmente pedidos, productos y usuarios a través de diversos criterios como nombre, fecha y estado, sin importar el uso de mayúsculas, minúsculas o acentos.

## 1.2 Documentos relacionados

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Descripción** | **URL ó nombre del archivo digital** |
| Documentación del proyecto |  | [Italia Pizza - Documento final.docx](https://uvmx-my.sharepoint.com/:w:/g/personal/zs21013882_estudiantes_uv_mx/EVj8FFzlAedIuaXuQWY7gEoBEEYzHZQJFVnt7CHOqUGtwg?e=aTF8Oe) |
| System Request |  | [System Request Italia pizza LIS.docx](https://uvmx-my.sharepoint.com/:w:/g/personal/zs21013862_estudiantes_uv_mx/EQnm9eY9F_RHuQL9zWKYuTUBBtoNI7TOnowqPcWe5eZ6Sg?e=wz73RT) |
| Diccionario de datos |  | [ItaliaPizza-DiccionarioDatos.xlsx](https://uvmx-my.sharepoint.com/:x:/g/personal/zs21013859_estudiantes_uv_mx/EQZqnixeu3tOhKwU_K72H9MButIv78FqhVm13sYf5tT4iQ?e=zWgVSf) |

## 1.3 Glosario

WPF (Windows Presentation Foundation): Un marco de desarrollo de software de Microsoft para la creación de aplicaciones de escritorio con interfaces de usuario avanzadas.

SQL Server: Un sistema de gestión de bases de datos relacional desarrollado por Microsoft para almacenar y recuperar datos solicitados por otras aplicaciones.

Entity Framework: Un marco de mapeo objeto-relacional (ORM) para .NET que permite a los desarrolladores trabajar con datos en forma de objetos de dominio específicos, sin tener que preocuparse por la complejidad de las bases de datos subyacentes.

Cliente-Servidor: Una arquitectura de red en la cual un servidor proporciona recursos o servicios a uno o más clientes.

Inventario: La lista detallada de los productos disponibles en stock, así como la cantidad de cada uno de ellos.

Corte de Caja: El proceso de revisar y contabilizar el dinero en efectivo y otros ingresos al final de un turno o día, para asegurar que coincida con las transacciones registradas.

Pedido: Una solicitud de productos realizada por un cliente, que puede incluir múltiples ítems.

PDF (Portable Document Format): Un formato de archivo utilizado para presentar documentos de manera independiente del software, hardware o sistema operativo en el que se visualizan.

Reporte: Un documento generado por el sistema que presenta datos específicos en un formato estructurado, como tablas o gráficos, a menudo utilizado para análisis y toma de decisiones.

Historial de Cortes de Caja: Registro mantenido en el sistema que guarda la información de todos los cortes de caja realizados durante un período determinado.

Edición de Pedidos: Funcionalidad que permite modificar los detalles de un pedido ya registrado, como agregar o quitar productos.

Estado del Pedido: Indicador del progreso de un pedido, que puede incluir estados como en espera de preparación, en preparación, listo, enviado, entregado, cancelado.

# 2. Alcance de las pruebas

## 2.1 Elementos de pruebas

|  |  |  |  |
| --- | --- | --- | --- |
| Módulos del sistema a probar | Objetivos de las pruebas | Detalle del orden de ejecución de los módulos | Responsabilidad de la prueba |
| Módulo de Usuarios | Validar el registro, edición y autenticación de usuarios. | 1. Módulo de Usuarios |  |
| Módulo de Productos | Validar la creación, edición y búsqueda de productos, así como la generación de reportes de inventario. | 2. Módulo de Productos |  |
| Módulo de Pedidos | Validar el registro, edición y búsqueda de pedidos, y la gestión completa del ciclo de vida de un pedido. | 3. Módulo de Pedidos |  |
| Módulo de Cocina | Validar la visualización y gestión de pedidos por preparar, y la actualización del estado de los pedidos. | 4. Módulo de Cocina |  |
| Módulo de Finanzas | Validar la gestión de ingresos y egresos, y la precisión en la realización de cortes de caja. | 5. Módulo de Finanzas |  |

## 2.2 Funcionalidades a probar

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de uso | Tipo de prueba | Módulo | #Casos de prueba a aplicar |
| Registrar cliente | Prueba Unitaria | Módulo de Usuarios |  |
| Buscar cliente | Prueba Unitaria | Módulo de Usuarios |  |
| Editar cliente | Prueba Unitaria | Módulo de Usuarios |  |
| Registrar dirección de cliente | Prueba Unitaria | Módulo de Usuarios |  |
| Editar dirección de cliente | Prueba Unitaria | Módulo de Usuarios |  |
| Registrar empleado | Prueba Unitaria | Módulo de Usuarios |  |
| Editar empleado | Prueba Unitaria | Módulo de Usuarios |  |
| Registrar proveedor | Prueba Unitaria | Módulo de Productos |  |
| Buscar proveedor | Prueba Unitaria | Módulo de Productos |  |
| Editar proveedor | Prueba Unitaria | Módulo de Productos |  |
| Registrar pedido de insumo | Prueba Unitaria | Módulo de Productos |  |
| Buscar pedido de insumos | Prueba Unitaria | Módulo de Productos |  |
| Editar estado de pedido de insumos | Prueba Unitaria | Módulo de Productos |  |
| Registrar nuevo producto | Prueba Unitaria | Módulo de Productos |  |
| Buscar producto | Prueba Unitaria | Módulo de Productos |  |
| Editar producto | Prueba Unitaria | Módulo de Productos |  |
| Generar reporte de inventario | Prueba de Integración | Módulo de Productos |  |
| Registrar pedido | Prueba Unitaria | Módulo de Pedidos |  |
| Buscar pedido | Prueba Unitaria | Módulo de Pedidos |  |
| Editar pedido | Prueba Unitaria | Módulo de Pedidos |  |
| Registrar nueva receta | Prueba Unitaria | Módulo de Productos |  |
| Buscar receta | Prueba Unitaria | Módulo de Productos |  |
| Editar receta | Prueba Unitaria | Módulo de Productos |  |
| Visualizar lista de pedidos por preparar | Prueba Unitaria | Módulo de Cocina |  |
| Registrar egreso de caja | Prueba Unitaria | Módulo de Finanzas |  |
| Realizar corte de caja | Prueba de Integración | Módulo de Finanzas |  |
| Registrar pedido a domicilio | Prueba de Integración | Módulo de Pedidos |  |
| Iniciar sesión | Prueba Unitaria | Módulo de Usuarios |  |
| Registrar insumo | Prueba Unitaria | Módulo de Productos |  |
| Editar insumo | Prueba Unitaria | Módulo de Productos |  |
| Buscar pedido a domicilio | Prueba Unitaria | Módulo de Pedidos |  |
| Editar pedido a domicilio | Prueba Unitaria | Módulo de Pedidos |  |

## 2.3 Funcionalidades a no probar

[Incluir si hay funcionalidades que no se probarán]

# 3. Criterios

## 3.2 Criterios de inicio

Antes de iniciar la ejecución de las pruebas, se deben cumplir los siguientes criterios:

* La documentación del diseño debe estar disponible para poder verificar los flujos y casos de uso.
* Disponibilidad de los ambientes de prueba, incluyendo hardware, software y bases de datos.
* Los datos deben ser validados para asegurar que sean representativos de escenarios reales de uso.

## 3.3 Bases de datos para las pruebas

Las bases de datos requeridas para las pruebas se encuentran en la siguiente ruta de acceso:

* Base de datos principal: **italiapizzadatabase.database.windows.net**

## 3.4 Criterios de aceptación o rechazo

Los criterios para evaluar la aceptación o rechazo del plan de pruebas incluyen:

* Debe incluir casos de prueba para cada caso de uso identificado.
* Los casos de prueba deben estar claramente descritos, incluyendo los pasos detallados a seguir, los datos de entrada necesarios, y los resultados esperados.
* Deben estar categorizados por tipo de prueba (unitarias, de integración, de sistema).
* El plan de pruebas debe ser revisado y aprobado por los principales interesados del proyecto, incluidos los equipos de desarrollo, pruebas, y el cliente o representante del negocio.

# 4. Recursos

## 4.1 Requerimientos de entorno – Hardware

* Bases de datos: italiapizzadatabase.database.windows.net

Equipos de cómputo para testers:

* Procesador: Intel Core i5 o equivalente.
* Memoria RAM: 8 GB.
* Almacenamiento: Disco duro con al menos 256 GB de capacidad.

Conectividad a la red:

* Velocidad: 144/144 Mbps.
* Adaptador de red: Intel(R) Wireless-AC 9560 160MHz.

## 4.2 Requerimientos de entorno – Software

Accesos a bases de datos de prueba: italiapizzadatabase.database.windows.net

MSTest fue la elección del equipo de pruebas para formar parte del entorno de software de pruebas. MSTest es el marco de pruebas unitarias integrado en el entorno de desarrollo de Microsoft y es ampliamente utilizado para la automatización de pruebas en aplicaciones .NET. La principal razón de que MSTest sea la elección es debido a su integración con Visual Studio, esta herramienta facilitará la creación, ejecución y gestión de pruebas dentro del mismo entorno de desarrollo. Además de que la familiaridad de la sintaxis de MSTest hace que los testers puedan escribir las pruebas de manera sencilla.

Instalación de software en computadoras asignadas a los Testers:

* Sistema Operativo: Windows 10 o versiones más recientes, que representan el entorno operativo principal de los usuarios finales.

## 4.3  Personal

El personal que realizará las pruebas serán los siguientes que además de ser el equipo de desarrollo, tendrán otras responsabilidades enfocadas a pruebas.

|  |  |
| --- | --- |
| **Nombre** | **Responsabilidad** |
| Aneth Michelle Tamariz Moreno | Tester, diseñador de pruebas y documentador |
| Sujey Alarcón Hernández | Tester, diseñador de pruebas y documentador |
| Eduardo Carrera Colorado | Tester, diseñador de pruebas y documentador |

# 5. Planificación y organización

## 5.1 Procedimientos para las pruebas

[Incluir si se ha implementado una metodología para las pruebas, probablemente una ágil para ir rápido]

## 5.2 Casos de pruebas

Los casos de prueba detallados se presentarán en un anexo adjunto al documento de plan de pruebas.

## 5.3 Cronograma

[Fechas para los tres tipos de prueba, entrega de documentos y correcciones]

## 5.4 Dependencias y riesgos

En relación con el proceso de pruebas de software se han detectado las siguientes dependencias y riesgos para el proyecto de Italia Pizza. Dado que contamos con una limitación de tiempo con un deadline inamovible, hemos enfocado los riesgos en las áreas más importantes del proyecto.

* La ejecución de las pruebas depende del estado de finalización de los componentes desarrollados.
* La disponibilidad de los recursos humanos es limitada, ya que solo contemos con tres testers que son cruciales para cumplir con el plazo del proyecto.
* Existe el riesgo de que los requisitos pueden cambiar durante el desarrollo, lo que podría causar retrasos.
* Podría existir un fallo en los servidores que interrumpa el proceso de pruebas.
* Las pruebas tienen un tiempo limitado ya que la fecha de entrega del proyecto es el día [Inserten día] así que esto genera un alto impacto en la planeación de las pruebas.
* Riesgo de que se detecten errores tardíos que representen un gran costo en el proyecto, ya sea de tiempo o esfuerzo con retrabajos.

Para estos casos se ha pensado que el plan de contingencia incluya la revisión y aprobación de cambios para evaluar su impacto, configurar un respaldo de la base de datos para poder probar en caso de que el servidor de Azure falle. Uno de los principales ideales que se han considerado es abordar los defectos encontrados de manera proactiva para asegurar una entrega exitosa del sistema.

# PARTE II. REPORTE DE PRUEBA

## 6. Actividades desarrolladas

## 6.1 Cambios

[Cambios que se realizaron como priorización de pruebas, enfoque en funcionales o adaptación a cambios en código]

## 6.2 Cronograma

[Hablar de si lo planeado se realizó]

# 7. Resultados

## 7.1 Principales incidencias

[¿Qué afectó a la ejecución de las pruebas?]

## 7.2 Cálculo de métricas

[Cuantificar la calidad de las pruebas]

**Fórmula:**

Cobertura de Pruebas con Casos de Uso (%)=(Casos de Uso Ejecutados / Casos de Uso Totales​)×100

## 7.3. Análisis de la métricas

[Cuánto sacamos según la métrica]

# 8. Conclusiones

## 8.1 Principales conclusiones

[Conclusiones generales]

## 8.2 Recomendaciones para la mejora del proceso

[Sugerencias para próximas pruebas de Italia Pizza]

# 9. Referencias

ISO/IEC/IEEE. (2021). *ISO/IEC/IEEE 29119-3 Software and systems engineering – Software testing- Part 3 Test documentation*. <https://www.iso.org/standard/79429.html>

Mikejo. (2023, 29 noviembre). *Ejecutar pruebas unitarias mediante el explorador de pruebas - Visual Studio (Windows)*. Microsoft Learn. <https://learn.microsoft.com/es-es/visualstudio/test/unit-test-basics?view=vs-2022>

# ANEXO A. Casos de prueba

[Resultados de los casos de prueba]