

PLAN DE PRUEBAS

Atlántico / Sovedidcost / Proyecto final

HISTÓRICO DE CAMBIOS

Fecha	Versión	Descripción	Autor
27-09-2021	1.0	Primera versión del plan de pruebas	Samly Guzman Maverick Fruto Jorge Gonzalez Emerson GonzalezR Roberto cahuana

ÍNDICE

1. INTRODUCCION	4
1.1 objetivos y tareas	4
1.1.1 objetivos	4
1.1.2 tareas	4
1.2 audiencia prevista	4
1.3 referencias	5
2. ALCANCE Y ENFOQUE	6
2.1 items a probar (funciones)	6
2.2 cuestiones de riesgo	6
2.3 características a probar	6
2.4 características que no se van a probar	6
2.5 enfoque (estrategia)	6
3. CRITERIOS DE TRANSICIÓN	7
3.1 criterios de entrada	7
3.2 criterios de salida	7
3.3 criterios de suspensión	7
3.4 criterios de reanudación	7
3.5 criterios de éxito y fallo	7
4. ESTRATEGIAS DE PRUEBA	8
5. PLANIFICACIÓN DE RECURSOS	9
5.1 Planificación	9
5.2 Recursos	9
5.2.1 Hardware	9
5.2.2 Software	9

1. INTRODUCCIÓN

Distribuciones y didácticos de la costa, es una empresa dedicada a la venta de útiles escolares a las distintas instituciones de la ciudad.

En base a esto, nosotros como aprendices Sena, decidimos desarrollar un aplicativo web, para dicha empresa; se diseñó este plan de pruebas, con el fin de evaluar y garantizar que los módulos desarrollados funcionen de manera correcta y cumplan con los requisitos de nuestro cliente.

1.1. OBJETIVOS Y TAREAS

1.1.1. Objetivos

- Implementar un plan de pruebas.
- Evaluar los módulos desarrollados por los aprendices.
- Garantizar un buen trabajo y el perfecto funcionamiento del sistema.

1.1.2. Tareas

- Elaborar el plan de pruebas
- Elaborar los casos de prueba.
- Elaborar el informe de dichas pruebas.
- Diseñar los casos de prueba
- Como se diseñaron los casos de prueba
- Escoger la estrategia para realizar la prueba.
- Determinar la técnica para realizar esa prueba.
- Registrar e resultado de las pruebas.

1.2. AUDIENCIA PREVISTA

Los instructores y el jefe de la empresa estarán auditando el proceso de pruebas que se le realicen a los distintos módulos del sistema. Con el fin de tener un resultado exitoso.

- Equipo de pruebas: Samly Guzman, Emerson Gonzalezrubio.
- Equipo de desarrollo: Roberto Cahuana, Jorge González
- Jefe de proyecto: Maverick Fruto.
- Grupos de aseguramiento de la calidad: Maverick Fruto, Samly Guzman, Emerson Gonzalezrubio.

1.3. REFERENCIAS

Lista todos los documentos que se han utilizado para crear este plan, los que se usarán en el desarrollo de casos de pruebas o durante la ejecución de pruebas. Estos se pueden listar en una tabla como la siguiente:

Documento	Autor	Versión	Localización
SRS	Grupo de proyecto	1.0	
Plan de calidad	Grupo de proyecto	1.0	
Plan de trabajo	Grupo de proyecto	1.0	
Pruebas de software	Grupo de proyecto	1.0	
Métricas	Grupo de proyecto	1.0	

2. ALCANCE Y ENFOQUE

2.1. ÍTEMS A PROBAR (FUNCIONES)

- Módulo de ventas.
- Módulo de login.
- Módulo de registro.

2.2. CUESTIONES DE RIESGO

Identifica qué software se ha de probar y cuáles son las áreas críticas, tales como:

- Entregable de un producto desarrollado por terceros.
- Capacidad de usar y entender una nueva herramientas
- Funciones extremadamente complejas
- Modificaciones de componentes con un histórico pasado de fallos
- Módulos o peticiones de cambio mal documentados

Hay algunos riesgos de software inherentes como la complejidad; éstos han de ser identificados.

Otra posible área de riesgo es el mal entendimiento de los requisitos originales. Esto puede ocurrir a nivel de gestión, usuario o desarrollador. Hay que tener en cuenta los requisitos que no están claros y los que no se pueden probar.

2.3. CARACTERÍSTICAS A PROBAR

- Login
- Base de datos
- Reportes.

2.4. CARACTERÍSTICAS QUE NO SE VAN A PROBAR

- Tiempo de respuesta del servidor y diseño de las interfaces.

2.5. ENFOQUE (ESTRATEGIA)

Para poder realizar las pruebas tomamos como estrategias las pruebas unitarias, estas se basan en comprobar el correcto funcionamiento de una unidad de código. Por ejemplo en diseño estructurado o en diseño funcional una función o un procedimiento, en diseño orientado a objetos una clase.

3. CRITERIOS DE TRANSICIÓN

A continuación se describen los criterios requeridos para las pruebas para poder pasar de un estado a otro.

3.1. CRITERIOS DE ENTRADA

Lista todos los criterios que se han de satisfacer para empezar la ejecución de las pruebas. Entre los posibles ítems que se pueden incluir están los siguientes:

- Aprobación del plan de pruebas
- Entorno de pruebas estable y preparado
- Casos de pruebas escritos y aprobados
- Herramientas de pruebas preparadas
- Recursos para las pruebas disponibles

3.2. CRITERIOS DE SALIDA

Lista todos los criterios que se han de satisfacer para que una fase de pruebas se de por finalizada. Entre los posibles ítems que se pueden incluir están:

- Completitud de los casos de pruebas
- Números y severidad de los defectos abiertos
- Paso de los objetivos de pruebas

3.3. CRITERIOS DE SUSPENSIÓN

-Fallas presentadas en el Hardware.

3.4. CRITERIOS DE REANUDACIÓN

-Corregir errores o fallas presentadas.

-Contar con los recursos disponibles.

3.5. CRITERIOS DE ÉXITO Y FALLO

Si más del 90% de las pruebas realizadas dieron como resultado un valor satisfactorio, se clasifica como éxito, de lo contrario si mas del 50% de las pruebas no mostraron buenos resultados sería una falla.

4. ESTRATEGIA DE PRUEBAS

Una Estrategia de prueba de software integra las técnicas de diseño de casos de prueba en una serie de pasos bien planificados que llevan a la construcción correcta del software. Es una parte fundamental del proceso de validación y verificación del software.

Sovedidcost, para la realización de estas pruebas, tomo como estrategias las pruebas de caja negra, que es una técnica de pruebas de software en la cual la funcionalidad se verifica sin tomar en cuenta la estructura interna de código, detalles de implementación o escenarios de ejecución internos en el software.

En las pruebas de caja negra, nos enfocamos solamente en las entradas y salidas del sistema, sin preocuparnos en tener conocimiento de la estructura interna del programa de software. Para obtener el detalle de cuáles deben ser esas entradas y salidas, nos basamos en los requerimientos de software y especificaciones funcionales.

Las pruebas de caja negra no utilizan ninguna información interna de los componentes de software o sistemas que se van a probar, sino que consideran el comportamiento del software desde el punto de vista de un observador externo, es decir, tal y como lo “viven” los usuarios del sistema.

Hay una secuencia de pasos a seguir media estandarizada para poder realizar este tipo de prueba de manera efectiva:

- Lo primero será un previo análisis de los requisitos y especificaciones del software.
- El tester diseñará una batería de entradas validas, también llamada escenario de prueba positiva, para verificar si el software las procesa correctamente. También se diseñan entradas no válidas (llamado escenario de prueba negativa) para comprobar si el software que se está probando es capaz de detectarlas y reaccionar antes estas entradas.
- Basándose en las entradas, el tester determina para cada una de estas las salidas esperadas correspondientes.
- Una vez que se tienen las entradas y su correspondiente salida, se diseña los casos de prueba.
- Se ejecutan esos casos de pruebas.
- El tester comprueba la salida que ha emitido el software con la salida esperada de los casos de prueba.
- Si la salida del software coincide con la salida esperada, el software hace lo que tiene que hacer para esa entrada. Pero si la salida del software no coincide con la salida esperada, hemos encontrado un defecto en el software. Lo que conllevará su posterior reparación.

5. PLANIFICACIÓN Y RECURSOS

5.1. PLANIFICACIÓN

- Aprobación del plan de pruebas
- Desarrollos de la lisara de casos de pruebas
- Desarrollo de los casos de pruebas
- Desarrollo de scripts de pruebas
- Preparación del entorno de pruebas
- Fechas de ejecución de pruebas

5.2. RECURSOS

5.2.1. Hardware

Para implantar nuestro sistema y realizar las pruebas, consideramos que se debería tener un pc, con las siguientes características:

- **CPU:** Intel Core i7-11700
- **RAM:** 16GB GSkill Trident Z 3600 Mhz
- **GPU:** NVIDIA GeForce 1060 6GB
- **PLACA BASE:** Gigabyte B560 Aorus Pro AX
- **ALMACENAMIENTO:** Samsung 970 EVO Plus 500GB
- **PSU:** Corsair RM V2 750W 80Plus
- **CAJA:** Nox Hummer MC USB 3.0

5.2.2. Software

Listado de los requisitos de software:

- Sistema operativo de Windows
- Microsoft office
- Base datos Mqsl
- Navegador google chrome
- Etc...

6. REVISION DEL PLAN DE PRUEBAS

ANEXOS	REVISOR	FECHA
Plan de casos de prueba	Samly Guzmán Emerson Gonzalezrubio	30/02/2021
Resultados de prueba	Samly Guzmán Emerson Gonzalezrubio	30/02/2021