

INSTITUTO TECNOLÓGICO DE MORELIA



Gestión de Proyectos de Software Plan de pruebas.

Prof.: Héctor Alonso Guzmán Gutiérrez

Alumnos: Miguel Ángel Domínguez Flores
Victor Daniel Romero Sheward

INDICE

1.- Objetivo:	3
2.- Estrategia de prueba:	3
2.1.- Orden de ejecución:	3
2.2.- Evaluación:	4
2.3.- Pruebas estructurales (caja blanca).	7
2.4.- Pruebas funcionales (caja negra).	7
2.5.- Pruebas de integración:	8
2.6.- Pruebas de aceptación:	8
2.7.- Responsables de la ejecución de pruebas unitarias:	8

1.- Objetivo:

La fase de pruebas tiene como propósito encontrar cualquier tipo de fallos, errores, malfuncionamientos o cualquier problema que implique algún tipo de insatisfacción o inconformidad por parte del cliente y corregirlo o hacer los cambios necesarios para entregar un producto de calidad que cumpla todas las expectativas.

Se contemplan pruebas unitarias que servirán para corroborar el correcto funcionamiento a nivel de código o de los módulos por separado, mientras que las pruebas de integración evaluarán que los módulos funcionen adecuadamente en conjunto.

2.- Estrategia de prueba:

2.1.- Orden de ejecución:

Se ejecutarán las pruebas para los casos de uso en el siguiente orden:

- Caso de uso de “Registro”.
- Caso de uso de “Ingreso”.
- Caso de uso de “Ver prestamos”.
- Caso de uso de “Ver material disponible”.
- Caso de uso de “Agregar usuario”.
- Caso de uso de “Eliminar usuario”.
- Caso de uso de “Agregar material”.
- Caso de uso de “Eliminar material”.
- Caso de uso de “Agregar un préstamo”.
- Caso de uso de “Modificar estado del material”.

2.2.- Evaluación:

Lista de verificación	
Caso de uso: Registro	
Campos obligatorios	
¿Se muestra con claridad cuáles son los campos obligatorios?	
¿Se muestra una alerta cuando se trata de enviar el registro con un campo obligatorio vacío?	
Campos con formato específico	
¿Se indica en que campos se requiere de una estructura específica?	
¿Se muestra una alerta cuando el campo en cuestión no cumple con el formato específico?	
¿Se impide el envío del formulario si el campo no tiene el formato adecuado?	
Mensajes	
¿Se muestra un mensaje cuando el formulario se ha enviado correctamente?	
¿Se muestra un mensaje o alerta cuando algún campo no cumple con las características requeridas?	

Lista de verificación	
Caso de uso: Ingreso	
Seguridad	
¿El texto del campo de contraseña aparece oculto?	
Mensajes	
¿Se muestra un mensaje o alerta cuando algún dato ingresado no es correcto?	

Lista de verificación	
Caso de uso: Ver préstamos	
Visualización	
¿Se muestran correctamente todos los campos correspondientes a los préstamos?	
¿Se muestra de manera correcta la cantidad de objetos prestados al usuario?	

Lista de verificación	
Caso de uso: Ver material disponible	
Visualización	
¿Se muestra el nombre del material en inventario correctamente?	
¿Se muestran las cantidades disponibles de cada material?	
¿Se muestra el total y cantidad prestada actualmente?	
¿Se muestran correctamente los detalles de cada material?	

Lista de verificación	
Caso de uso: Agregar usuario	
Campos obligatorios	
¿Se muestra con claridad cuáles son los campos obligatorios?	
¿Se muestra una alerta cuando se trata de realizar el registro con un campo obligatorio vacío?	
Campos con formato específico	
¿Se indica en que campos se requiere de un formato específico?	
¿Se muestra una alerta cuando el campo en cuestión no cumple con el formato específico?	
¿Se impide el envío del formulario si el campo no tiene el formato adecuado?	
Mensajes	
¿Se muestra un mensaje cuando el usuario se ha agregado correctamente?	
¿Se muestra un mensaje o alerta cuando algún campo no cumple con las características requeridas?	

Lista de verificación	
Caso de uso: Eliminar usuario	
Alertas y mensajes	
¿Se muestra una alerta de confirmación antes de eliminar un usuario?	
¿Se muestra un mensaje cuando un usuario ha sido eliminado correctamente?	

Lista de verificación	
Caso de uso: Agregar material	
Visualización	
¿Se muestran correctamente el campo de nombre del material?	
¿Se muestra correctamente el campo de descripción del producto?	
¿Se muestra correctamente el campo de cantidad del producto a ingresar?	
Mensajes	
¿Se muestra un mensaje cuando el producto se ha agregado correctamente?	
¿Se muestra un mensaje o alerta cuando algún campo no se llenó correctamente?	

Lista de verificación	
Caso de uso: Eliminar material	
Alertas y mensajes	
¿Se muestra una alerta de verificación antes de eliminar un material?	
¿Se muestra un mensaje cuando el material ha sido eliminado satisfactoriamente?	

Lista de verificación	
Caso de uso: Agregar un préstamo	
Visualización	
¿Se muestra con claridad todos los campos?	
Numeración	
¿Se muestra correctamente la cantidad de material disponible?	
¿Al agregar una cantidad de material al préstamo, esta se resta del material disponible?	
¿Las cantidades son congruentes y siempre positivas?	
Mensajes	
¿Se muestra un mensaje cuando el préstamo se registra correctamente?	
¿Se muestra un mensaje o alerta cuando algún campo no cumple con las características requeridas?	

Lista de verificación	
Caso de uso: Modificar estado del material	
Visualización	
¿Se muestra correctamente el estado actual del material?	
¿Se muestran correctamente las cantidades de material?	
Mensajes	
¿Se muestra un mensaje cuando se modifica exitosamente el estado de un material específico?	

2.3.- Pruebas estructurales (caja blanca).

En este tipo de prueba el objetivo es recorrer en detalle el código y verificar su correcta ejecución en cada una de las circunstancias posibles de acuerdo al funcionamiento esperado del programa. Para ello se llevarán a cabo los siguientes pasos:

Paso 1: Se organizarán los elementos del código a revisar de acuerdo a su complejidad.

Paso 2: Se asignarán estos elementos a los equipos de revisión, conformados de acuerdo a la complejidad del elemento a revisar.

Paso 3: Se verificará cada posible resultado de la ejecución de los elementos.

Paso 4: Se realizará un reporte de las observaciones de cada prueba.

Paso 5: El administrador de pruebas evaluará los reportes obtenidos y los comparará con los resultados esperados.

Paso 6: De ser necesario, se asignarán equipos de corrección para cada elemento que no funcione de acuerdo a lo esperado.

Paso 7: Se hará una revisión que verifique que la corrección ha resuelto satisfactoriamente el problema inicial, y que no ha causado problemas nuevos.

Paso 8: Se generará un documento donde se resuma el proceso y los cambios aplicados.

2.4.- Pruebas funcionales (caja negra).

Estas pruebas se llevan a cabo sobre la interfaz del programa para verificar que funcione adecuadamente desde el punto de vista del usuario, es decir, de manera externa al código.

Con ello se detectarán y corregirán cualquier tipo de anomalías propias de la interfaz de usuario, su estructura y los elementos con los que interactúa directamente.

De este modo, se demostrará lo siguiente:

- Que el software opera correctamente.
- Que las entradas funcionan de manera correcta.
- Que las salidas son correctas.
- La integridad externa del software.
- Que las operaciones son congruentes.
- Que la información no se modifica de manera indebida.
- Que ciertas funcionalidades correspondan al tipo de usuario correcto.

Para ello se realizarán las siguientes pruebas de la funcionalidad del sistema:

Paso 1: Realizar algunos registros con diferentes datos esperados y erróneos para detectar si acepta datos incorrectos de algún tipo.

Paso 2: Ingresar como usuario registrado y verificar las secciones y opciones a las que se tiene acceso.

Paso 3: Navegar entre secciones y opciones, verificando que se visualicen de forma correcta y con sus elementos respectivos.

Paso 4: Realizar un reporte de las pruebas, especificando la sección y, de haberlo, el problema o malfuncionamiento observado.

Paso 5: El administrador de pruebas revisará los reportes y asignará equipos de corrección en caso de ser necesario.

Paso 6: Se harán las correcciones y se ejecutarán las mismas pruebas con los elementos nuevos.

Paso 7: Una vez se haya llegado a un reporte completamente satisfactorio, se dará por aprobada esta fase.

2.5.- Pruebas de integración:

Estas pruebas evaluarán el correcto funcionamiento de los módulos una vez implementados en conjunto con su respectiva estructura. Para ello se evaluará cada modulo de manera individual y posteriormente se unirá de manera incremental a la estructura principal, donde se llevará a cabo nuevamente una prueba de su funcionamiento para verificar que no existan conflictos o errores derivados de la implementación conjunta de otros módulos.

2.6.- Pruebas de aceptación:

Estas pruebas se realizarán con el cliente y algunos usuarios ajenos al desarrollo simulando su ambiente de operación ordinario. De este modo se espera encontrar algún detalle que haya escapado a los filtros anteriores y evaluar desde un punto de vista ajeno, y con cierto enfoque a la experiencia del usuario.

2.7.- Responsables de la ejecución de pruebas unitarias:

Para las pruebas unitarias serán asignados equipos o personal ajeno al desarrollo del módulo a evaluar, de modo que se tenga otra perspectiva y las pruebas no sean basadas en el conocimiento previo del funcionamiento de los elementos.