

Actividad | 3 | Plan de pruebas y Reglas para el
Versionado e Incidencias

Aseguramiento de la Calidad

Ingeniería en Desarrollo de
Software

TUTOR: Jessica Hernández Romero

ALUMNO: Adriana Esteban López

FECHA: 19 de mayo de 2025

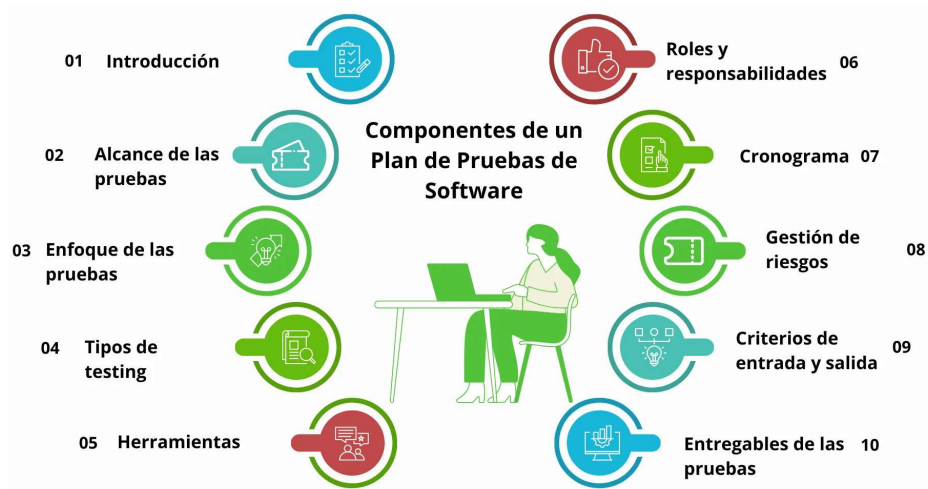
INDICE

INTRODUCCIÓN	03
DESCRIPCIÓN	05
JUSTIFICACIÓN	07
DESARROLLO	08
CONCLUSIÓN	11
REFERENCIAS	12

INTRODUCCIÓN

Dentro de la elaboración de cualquier proyecto se requiere de un seguimiento, y el desarrollado del software no es la excepción y para ello contamos con una herramientas llamada ***plan de pruebas*** el cual consiste en un documento en el cual se define cómo y cuándo se estarán llevando a cabo el proceso de pruebas.

En la siguiente imagen se puede apreciar los componentes de un Plan de Pruebas:



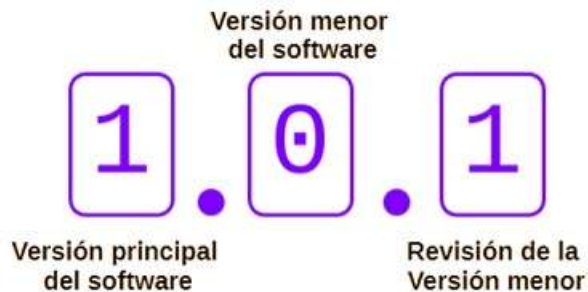
Algunas sugerencias para crear un buen plan de pruebas es:

1. Fomentar el trabajo en equipo.
2. Casos de pruebas reales en base a lo que el cliente solicita en sus requerimientos.
3. Actualización constante del plan de pruebas para la retroalimentación al equipo de trabajo.

Una vez que se tiene el plan de pruebas y que estas se están ejecutando conforme a lo planeado, tenemos también que ir viendo las “versiones” que se van generando del sistema, conforme lo vamos codificando y probando cada módulo, para ello contamos con la herramienta de ***Versionado e Incidencias***.

Este control tiene como objetivo el tener un control de las versiones que van surgiendo del software que estamos desarrollando conforme vamos implementando los módulos y/o actualizaciones de los mismo, para garantizar la funcionalidad de cada uno y en conjunto.

Pero ¿cómo se deciden las versiones de un software?, aunque existen varias propuestas a nivel mundial, una de las más utilizadas es identificarlas a través de un número, en la cual generalmente se utilizan 3 dígitos:



La **versión principal** estará cambiando cuando se realice una modificación que incluya una nueva función de mucha importancia en el sistema, la **versión menor** estará cambiando cuando se realiza una modificación menor al sistema y que no representa un cambio significativo en la operación del sistema y finalmente la **versión (revisión)** es cada vez que se realiza la entrega del sistema.

En el desarrollo de esta actividad estaremos plasmando un plan de pruebas y control de versiones en base al ejemplo que se ha manejado en el contexto desde la actividad 1.

DESCRIPCIÓN

Contextualización:

La compañía Cajas y Derivados S.A. de C.V. ha crecido tanto que necesita llevar su control de inventario y clientes en un sistema multiplataforma en línea. De la misma manera, se pretende que este control se pueda acceder desde cualquier dispositivo con acceso web.

De acuerdo a la actividad 1 de la compañía “Cajas y Derivados S.A. de C.V.” es necesario reconocer la importancia de los casos de pruebas y la matriz de prueba con el objetivo de escenificar el software con tal de medir la funcionalidad del mismo.

Requerimientos técnicos:

- El sistema será desarrollado con los lenguajes HTML5, CSS, Python y un patrón de arquitectura MVC.
- El modelo encargado de conectarse a la base de datos será postgresQL. Se pide que el programa presente las siguientes características:
- Se le solicitará al personal ingresar su número de empleado y contraseña, los cuales serán consultados en la base de datos. De esta manera, si el usuario no existe o la contraseña no corresponde a este, se le informará mediante un mensaje de error.
- Toda la información presentada por la aplicación deberá mostrarse conforme a los formatos corporativos definidos por la organización.
- El sistema deberá permitir tener dos perfiles: Administrador y Empleado.
- El sistema deberá permitir al administrador dar de alta a los diferentes empleados; en caso de que ya exista un empleado y el administrador quiera volver a agregarlo, el sistema deberá mostrar un error.
- El sistema deberá permitir a los empleados dar de alta los productos del inventario; en caso de que el producto ya exista, deberá mostrar un mensaje de error.
- El sistema proporcionará la opción de generación de reportes de inventarios, clientes y empleados.

Requerimientos funcionales:

- El sistema debe ser veloz.
- El sistema debe ser fácil de entender.
- El sistema debe tener información encriptada.
- La información debe ser respaldada cada 7 días.

Actividad

- Crear un plan de pruebas integral, incluir: cronograma y definición de los roles del equipo de desarrollo. Además especificar los estándares.
- Establecer las reglas para el versionado y manejo correcto de incidencias.

JUSTIFICACIÓN

La importancia del desarrollo e implementación de un plan de pruebas es la de garantizar que todos los aspectos de un software serán revisados bajo los criterios que garanticen que el sistema estará cumpliendo con lo que el cliente solicito.

Por otro lado es importante también estar llevando un control de las versiones que se van realizando del sistema que se esta desarrollando, ya que podemos cometer el error de estar guardando copias que en algún momento es poco útil y corremos el riesgo de no implementar todas las mejoras a la vez en el sistema final; además de que con ello podemos llevar un control de las incidencias que se van presentando y por las cuales vamos mejorando el código del sistema y por ende van surgiendo las versiones.

Como podemos ver el versionado del software es un proceso a través del cual vamos a poder establecer un “nombre” al software para poder indicar el nivel de desarrollado en que se encuentra el sistema que estamos desarrollando.

DESARROLLO

Plan de Pruebas

Introducción: El diseño de este plan de pruebas tiene como objetivo garantizar que el sistema de Inventarios de la Compañía Cajas y Derivados S.A. de C.V cumpla con los requerimientos que se requieren y las condicionantes para operar el sistema.

Alcance de las pruebas: Las pruebas que se propongan en este plan estarán midiendo el comportamiento de la Base de Datos, que el sistema sea operable en los diferentes sistemas operativos que el cliente requiere y con la arquitectura MVC.

Se estarán evaluando los módulos de *Login*, *Altas* y *Registro de productos en Inventario*.

Objetivos de las pruebas:

- Validar que el usuario que inicie sesión sea un empleado o directivo, a través del usuario (número de empleado) y contraseña sean correctos (estén dentro de la Base de Datos).
- Validar que solo el usuario con estatus de directivo pueda dar el alta de un empleado, y que en caso de que este ya exista, mandar el mensaje de error correspondiente, o permitir guardar los datos del nuevo empleado
- Validar que el empleado pueda ingresar al inventario un nuevo artículo, y en caso de ya existir en la Base de Datos emitir el mensaje correspondiente.

Tipos de Pruebas: Las pruebas que estaremos utilizando son:

- Pruebas Unitarias: Evaluar el Login, Alta, y Registro de productos en Inventario.
- Pruebas de Integración: Correcto funcionamiento entre los módulos a evaluar con las pruebas unitarias.
- Pruebas de Sistema: Funcionalidad completa del sistema
- Pruebas de aceptación: que el sistema cumpla con los requisitos que solicito el cliente.

Herramientas: Se hará uso de HTML5, CSS, Python para el desarrollo del sistema, así como también de SQL para el desarrollo de la Base de Datos.

Roles y responsabilidades: Los roles que estarán participando en este desarrollo serán:

- Líder del Proyecto: Estará coordinando al equipo de principio a fin en el desarrollo de todo el sistema.
- Analista de Sistemas: Se encargará de plasmar de manera clara y precisa los requerimientos que el cliente requiere que tenga el sistema de Inventarios.
- Desarrollador de Base de Datos: Diseñará la Base de Datos.
- Desarrollador de Software: Estará realizando la escritura del código que se requiera para el desarrollo del Sistema de Inventarios, y en este caso se requiere de expertos, uno para cada lenguaje de programación.
- Probador de Sistemas: Se encargará de aplicar las pruebas que se establezcan para garantizar que el sistema cumpla con los requerimientos que halla establecido el cliente.
- Administrador de Sistemas: Se estará encargando de implementar el sistema en conjunto y funcionando tal cual el cliente lo solicito.

Cronograma de pruebas:

Cronograma de Actividades										
Fases	Actividades	Rol	Semanas							
			1	2	3	4	5	6	7	8
Análisis de requisitos	Requerimientos técnicos	Analista de Sistemas								
	Requerimientos funcionales									
	Definición de restricciones									
Planificación	Asignación de Roles	Líder del Proyecto								
	Diseño de la Base de Datos en SQL	Desarrollador de Base de Datos								
	Diseño del sistema en los lenguajes HTML5, CSS, Python con una arquitectura MVC	Desarrollador de Software								
	Diseño de Casos de Pruebas	Probador de sistemas								
Ejecución	Implementación de la Base de Datos	Administrador de Sistemas								
	Implementación del Sistema									
	Asignación de perfiles en sistema									
	Implementación de las pruebas									
Seguimiento y Control	Revisión de resultados de los casos de pruebas	Administrador de Sistemas								
	Resolución de problemas									
	Pruebas finales									
Cierre	Entrega del sistema funcionando	Líder del Proyecto								

Gestión de riesgos: En base a los requerimientos y condicionantes que establezcan se determinará cuáles son los riesgos que se pueden presentar y cuáles serían las soluciones.

Resultados de pruebas: Una vez realizadas las pruebas, e identificados los riesgos que se pueden presentar así como las soluciones, establecer por escrito como actuar ante cada situación.

Documentación de las pruebas: Se estará realizando un documento en el cual se plasmen las pruebas realizadas y cual fue el resultado de ellas.

Control de versiones:

Control de Versiones del Sistema de Inventarios			
Versión	Fecha	Descripción	Autor
1.0.0	05 de mayo 2025	Diseño de Base de Datos	Desarrollador de Base de Datos
1.1.0	12 de mayo de 2025	Diseño de Sistema (código)	Desarrollador de Base de Software
1.1.1	19 de mayo de 2025	Integración del Módulo de perfiles	
1.1.2		Integración del Módulo de Login	
1.1.3		Integración del Módulo de Altas	
1.1.4		Integración del Módulo de Registro de Productos en Inventario	
1.2.0	26 de mayo de 2025	Integración de todos los módulos en el sistema	Líder del Proyecto
1.2.1	02 de junio de 2025	Entrega del Sistema	

CONCLUSIÓN

Con el desarrollado de esta actividad podemos concluir que el desarrollado del plan de pruebas es una actividad de gran importancia dentro del desarrollado del software, ya que este nos brindará una organización, ejecución y seguimiento de las pruebas establecidas para garantizar la calidad del sistema desarrollado, en cuanto a que estará cumpliendo los requisitos solicitados por el cliente.

Es decir, la ejecución del plan de pruebas de manera adecuada y correcta es vital e indispensable para el éxito del desarrollado del software, para que al momento del lanzamiento de dicho sistema no haya situaciones que comprometan su funcionamiento.

Por otro lado, aunque no hay una regla completamente estandarizada de como estar dando nomenclatura a las versiones, consideramos que la más viable y a la que estamos acostumbrados es en la que se utilizan número, ya que puede ser más fácil de utilizar.

A lo largo del desarrollo de estas actividades, podemos concluir que el proceso de revisión del sistema desde el punto de vista del usuario final en cuanto a que realiza o no las funciones que establece el cliente, también es una actividad esencial y de mucha importancia, al igual que la parte de planificación, implementación y revisión del sistema desde el código; con la finalidad de garantizar que al final se entregue un sistema de calidad, acorde a los requerimientos que establece el cliente.

Se agrega dicha actividad a la plataforma de GitHub a través del siguiente link:

<https://github.com/22HADRIA/Aseguramiento-de-la-Calidad>

REFERENCIAS

Plan de pruebas de software: 5 elementos clave

<https://www.testingit.com.mx/blog/plan-de-pruebas-de-software>

Guía completa para elaborar un plan de pruebas de software

<https://qalified.com/es/blog/plan-pruebas-software/>

Cómo crear un plan de pruebas (pasos, ejemplos y platillas)

<https://www.testrail.com/blog/create-a-test-plan/#test-plan-faqs-6>

¿Qué es el Control de Versiones?

<https://learn.microsoft.com/es-es/devops/develop/git/what-is-version-control>

¿Cómo se deciden las versiones del software?

<https://ed.team/blog/como-se-deciden-las-versiones-del-software>