

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

Aseguramiento de la Calidad.

Ingeniería en Desarrollo de Software.



TUTOR: Jessica Hernandez Romero.

ALUMNO: Ramón Ernesto Valdez Felix.

FECHA: 12/05/2024.

Introducción.	3
Descripción.	3
Justificación.	4
Desarrollo:	5
Plan de Pruebas Integrales.	5
Reglas para el versionado e incidencias.	7
Conclusion.	10
Referencias.	11

Introducción.

En la actividad final de la materia de aseguramiento de la calidad, nos plantea realizar la documentación de Plan de pruebas y Reglas para el Versionado e Incidencias de la aplicación web multiplataformas de la compañía Cajas y Derivados S.A. de C.V., ya que ha tenido un crecimiento continuo que requiere y necesita llevar un control de su inventario y clientes de su sistema de multiplataforma que se encuentra en línea. de igual manera pretende que este control se lleve a cabo desde cualquier dispositivo que tenga acceso web seguro. Este deberá ser digerible o muy intuitivo para cualquier tipo de usuario, esto para que sea atractivo y el usuario quede fascinado con su usabilidad, veloz como una gacela, seguro como bóveda y adicional se deberá contar con un resguardo continuo de la información o mejor conocidos como respaldo de seguridad de toda la información que se contenga en la aplicación esto como buena práctica y como punto de recuperación en caso de desastres.

Descripción.

En la actividad final de la materia de aseguramiento de la calidad, la compañía Cajas y Derivados S.A. de C.V., solicita nuestro servicio para ayudar con el Plan de pruebas y Reglas para el Versionado e Incidencias del desarrollo que está requiriendo para llevar el control de su inventario y clientes, utilizando la información de las actividades anteriores donde la aplicación con la que requerimos trabajar con el Plan de pruebas y Reglas para el Versionado e Incidencias tiene como objetivo de escenificar el software con tal de medir la funcionalidad del mismo, aplicación multiplataformas donde cualquier dispositivo con acceso web lo pueda consumir de manera fácil y rápida, Adicional debe de ser segura contando su cifrado o encriptación del canal de comunicación de la aplicación con sus dispositivos a consumir y deberá contar con un procedimiento continuo de respaldos para cualquier contingencia no prevista y contar un plan de recuperación de desastres para actuar de manera más rápida y eficaces.

Justificación.

En esta actividad trabajaremos con la documentación proporcionada por el docente de la materia de aseguramiento de la calidad se realizará la actividad de Plan de pruebas y Reglas para el Versionado e Incidencias, utilizando todos los punto de la documentación de las actividades anteriores que se utilizará como información adicional a este documento como los requerimientos técnico, características del programa y requerimientos funcionales con esta información realizaremos las actividades de la actividad uno de la materia:

- 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.
- Diseñar casos de prueba para el aplicativo.
- Diseñar matriz de prueba para el aplicativo.
- 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 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.

5

- 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.
- 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.

Desarrollo:

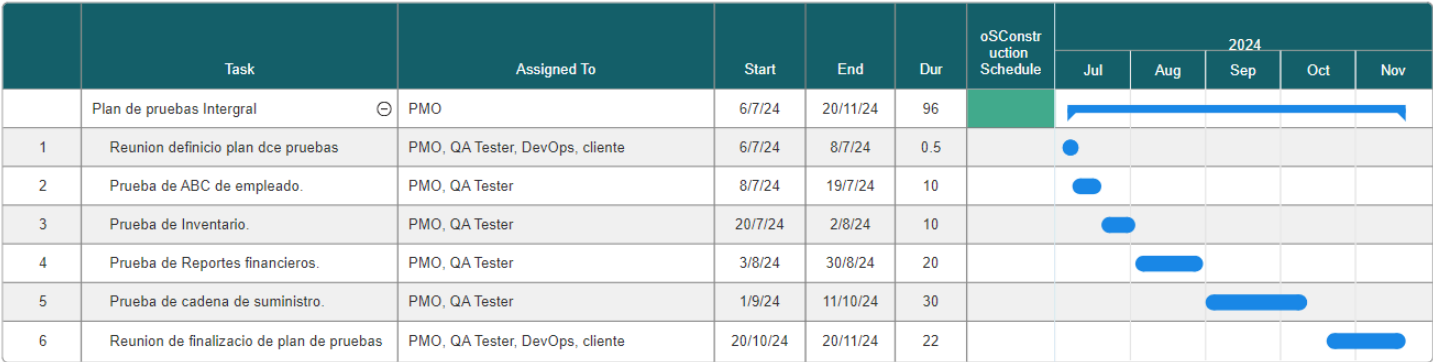
En la siguiente actividad del documento realizaremos el desarrollo de la actividad final donde redactamos los puntos de casos de Plan de pruebas y Reglas para el Versionado e Incidencias para la aplicación web multifuncional de la compañía Cajas y Derivados S.A. de C.V. ya que el crecimiento exponencial de la empresa requiere nuestro apoyo para la implementación correcta de la aplicación multiplataformas online, adicional se anexa el link de la documentación en el link siguiente.

Link: [GitHub](#)

Plan de Pruebas Integrales.

El objetivo de esta actividad de un plan de pruebas integrales es proporcionar una guía detallada para todas las actividades de prueba necesarias que realizaremos en la aplicación web de la empresa Cajas y Derivados S.A. de C.V esto para garantizar la calidad del software. Este documento sirve como referencia para el equipo de pruebas, asegurando que todos estén alineados en cuanto a los objetivos y

las actividades de prueba. Y anexamos estándares internacionales a los cuales deben de estar apegados los desarrollos de las aplicaciones.



En el cronograma presentado los roles que se muestra en la siguientes tabla:

Roles: Responsabilidades:	
PMO	Es la entidad dentro de una organización que se encarga de estandarizar y optimizar los procesos relacionados con la gestión de proyectos
QA Tester	Su principal responsabilidad es evaluar el software en desarrollo para identificar defectos, errores o problemas que puedan afectar la funcionalidad, la usabilidad o la seguridad del producto final.
DevOps	Equipos de desarrollo y operaciones, fomentando la colaboración, la automatización y la entrega continua para mejorar la calidad, la velocidad y la eficiencia en el desarrollo y la entrega de software.
Cliente	Solicitante del servicio de aseguramiento de calidad para el desarrollo que requiere queso implementado en su empresa.

Estándares:	
ISO 29119	Es una serie de normas internacionales que proporciona pautas y estándares

	para la gestión de pruebas de software. Estas normas están diseñadas para ayudar a las organizaciones a mejorar la calidad de su proceso de pruebas y a garantizar que las pruebas se realicen de manera efectiva y eficiente
ISO 9001	Implica establecer un sistema de gestión de calidad eficaz que se centre en satisfacer las necesidades y expectativas del cliente, mejorar continuamente los procesos de desarrollo de software y tomar decisiones basadas en datos y evidencia.
IEEE 829	Es un estándar utilizado en la industria del software que describe el formato y el contenido de los documentos de prueba de software. Está diseñado para proporcionar una estructura coherente y consistente para la documentación de pruebas en proyectos de desarrollo de software. Este estándar define una serie de documentos de prueba que cubren diferentes aspectos del proceso de prueba.

Reglas para el versionado e incidencias.

En la actividad siguiente hablaremos sobre las reglas para el versionado e incidencias, y comenzaremos estableciendo las reglas del versionado que nos dará un entorno claro siguiendo estas reglas básicas de versionado, puedes mantener un proceso de desarrollo de software estructurado, controlado y orientado a la calidad.

Reglas para el Versionado:

Versión consistente:	Cada vez que se realice una reunión de definición del plan de pruebas, se deberá asignar una versión específica al software en desarrollo. Esto garantizará que cada iteración del plan tenga una versión única y consistente asociada.
Registro de versiones en reuniones:	En cada reunión de definición del plan de pruebas, se debe documentar la versión asignada al software. Esto asegurará que haya un registro claro y actualizado de todas las versiones del software en desarrollo.
Control de cambios en el código:	Cada vez que se realice una reunión de definición del plan de pruebas, se deberá verificar si hay cambios en el código fuente del software. En caso afirmativo, se deberá registrar y documentar adecuadamente dichos cambios para mantener un seguimiento preciso del historial de versiones.

Versiones por estabilidad.

Las opciones que tenemos para esto son: Alpha, Beta.

Alpha es una versión inestable que es muy probable que tenga muchas opciones que mejorar, pero queremos que sea probada para encontrar errores y poder poner a prueba funcionalidades, en la mayoría de los casos podemos decir que está casi listo el producto. Ejemplo: 1.0Alpha, 1.0a1, 1.0a2.

Beta es una versión más estable que Alpha en la que contamos con el producto en su totalidad, y se desea realizar pruebas de rendimiento, usabilidad y funcionamiento de algunos módulos para ver cómo funciona bajo un ambiente no tan controlado. Aquí aparece el nombre de Beta Tester que escuchamos mucho en el mundo del software. Ejemplo: 2.0Beta, 2.0b, 2.0b1

El siguiente paso es RC (Release Candidate), que es el último toque fino del software antes de salir y después de pasar por Beta. Ejemplo: 3.0-RC o también 3.0-RC1.

Reglas para la gestión de incidencias, el equipo de desarrollo puede abordar de manera efectiva y eficiente los problemas que surjan durante el desarrollo de software, garantizando así la calidad y la satisfacción del cliente.

Reglas para la Gestión de Incidencias:

Registro de incidencias en reuniones:

Durante cada reunión de definición del plan de pruebas, se debe discutir y registrar cualquier incidencia o problema identificado en versiones anteriores del software. Esto garantizará que todas las incidencias sean abordadas y priorizadas adecuadamente.

Asignación de responsabilidades:	Cada vez que se identifique una incidencia durante una reunión de definición del plan de pruebas, se deberá asignar a un miembro del equipo la responsabilidad de abordar y resolver dicha incidencia. Esto garantizará que las incidencias sean tratadas de manera oportuna y eficiente.
Seguimiento de incidencias resueltas:	Después de cada reunión de definición del plan de pruebas, se debe hacer un seguimiento de las incidencias identificadas y su estado de resolución. Esto ayudará a garantizar que todas las incidencias sean abordadas y resueltas adecuadamente antes de la siguiente iteración del plan de pruebas.

Conclusion.

En conclusión: tanto el plan de pruebas como las reglas para el versionado e incidencias son aspectos críticos en el desarrollo de software que contribuyen significativamente a la calidad, estabilidad y eficiencia del proceso de desarrollo. Un enfoque estructurado y organizado en ambos aspectos es esencial para garantizar un producto final de alta calidad que cumpla con las expectativas del cliente y los requisitos del mercado. Por dicho motivo la app multiplataforma de acceso web de la compañía Cajas y Derivados S.A. de C.V. debe cumplir con todos estos aspectos de aseguramiento de la calidad para ser un producto confiable,veloz y segura tanto para su dueño como para sus consumidores llevándolo a la

evolución de la forma de trabajo tradicional y embarcando a una mejora continua de sus app y procesos empresariales. adicional a lo anterior las reglas para el versionado e incidencias son elementos fundamentales en el proceso de desarrollo de software que contribuyen a la calidad, la eficiencia y el éxito general del proyecto. Un enfoque estructurado y organizado en ambos aspectos es esencial para garantizar un producto final de alta calidad que cumpla con las expectativas del cliente y los requisitos del mercado.

.

Referencias.

GitHub: Let's build from here. (n.d.)

¿Cómo se deciden las versiones del software? (n.d.). ..Ed.team. Retrieved May 13, 2024, from <https://ed.team/blog/como-se-deciden-las-versiones-del-software>