



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

Nombre del curso

Ingeniería en Desarrollo de Software



TUTOR: Jessica Hernández.

ALUMNO: Uziel de Jesús López Ornelas.

FECHA: 15 de Mayo Del 2025.

Tabla de Contenido

Introducción	1
Descripción	1
Justificación	2
Desarrollo	3
Plan de Pruebas Integral.....	3
Reglas para el Versionado e Incidencias.....	5
Conclusión	5
Referencias	6

Introducción

Estamos por terminar esta materia y en este proyecto final tenemos que crear un cronograma en el que se alisten y estructuran todas las actividades que se tienen que realizar, colocar responsables, fechas en las que se estaría terminando las actividades y por supuesto los estándares necesarios para que el proyecto se concluya de la mejor manera posible y detectar todos los errores y defectos que podemos encontrar en el proceso de esta parte de la actividad, también nos pide que se realice un versionado y manejo correcto de errores, esto es simple, si se tiene una versión “1.1” y esta presenta una nueva función pero viene con ella un pequeño error, la otra actualización o versión pudiera ser la “1.2” que ofrece mejoras y correcciones a la anterior y por ende mejora su calidad. Estas dos actividades serán las principales a realizar para el proyecto final, ya que es con lo que cerraremos para complementar los temas y conceptos de esta materia.

Descripción

Definiremos que es un caso de prueba para entender la importancia de ordenar parámetros en los proyectos de este tipo, pues esto es un conjunto de acciones con resultados y salidas ya previstas, es decir que se tiene un cierto conocimiento de cómo este tendría su resultado final, a la vez estos tienen sus diferentes componentes que lo representan:

- Propósito de la prueba o descripción del requisito que se está probando en ese momento.
- El método o forma en la que se probará.
- La versión o configuración de la prueba, el hardware, el software, el SO, los archivos, entre otros componentes que complementan a la prueba.
- Los resultados y acciones esperados, o entradas y salidas.
- La documentación de la prueba y sus anexos.

Un caso de prueba debe de contar con distintos factores que representan la calidad en la prueba y son fundamentales que se tomen en cuenta:

- Debe ser exacto, es decir que su descripción indique que se puede probar.
- Debe tener solo los pasos o campos necesarios para probarlo, no generar de sobra que pueda ser redundante por lo que esto es económico.
- La confianza y que este sea repetible hace que un experimento este controlado cada vez que se llega al mismo resultado en la ejecución.

- Rastreadable, esto indica que se tiene que saber los requisitos del caso de uso se aprueban.
- Estos deben de cumplir con estándares para que se acoplen a lo demás, es decir que sea medible.

Un caso de prueba efectivo y que no demuestra complicaciones genera en si resultados más confiables, que ayudan a mejorar su rendimiento y reducir costos:

- La productividad.
- La capacidad de prueba.
- La programación de la fiabilidad.

Son conceptos que se tienen relacionados a que una prueba salga satisfactoria en todos los ámbitos.

Justificación

Se vio anteriormente lo que son los casos de prueba de manera manual, pero también existen pruebas que son completamente automatizadas y que en ocasiones ayudan a reducir el estrés y trabajo en las pruebas, aplicaciones como estas son las siguientes:

- Selenium.
- Watir.
- SoapUI.
- Junit.
- TestNG.
- Nunit.
- Testoob.

En las pruebas como ya se ha dicho se pueden encontrar errores, defectos o incidencias que afectan al proyecto en su ejecución del mismo, una incidencia se define como un acceso, intento de acceso, uso o divulgación no autorizada de información, en pocas palabras es una ocurrencia de un suceso que requiere investigación. Este al igual que otros, maneja un ciclo de vida que es necesario en lo que respecta a cualquier situación ya que es universal:

- Fase inicial que es la prevención y preparación, la detección y el pre análisis.
- La contención, erradicación y recuperación.
- Recuperación del incidente.

- Actividad después del incidente (La reflexión y documentación).
- Todo esto nos ayuda a actuar en caso de que ocurra otra incidencia parecida o completamente nueva, actuar rápidamente y de acuerdo a los parámetros ayuda a que no se salga de control.

Desarrollo

Plan de Pruebas Integral

[illegible]

[illegible]

Reglas para el Versionado e Incidencias

Control de Versionado					
Versión	Fecha de Inicio	Fecha Final	Descripción	Encargado	Puesto
v 0.5	14/05/2025	16/05/2025	Pruebas en el diseño de la BD en postgresQL.	Juan Ramírez	Programador
v 0.5	17/05/2025	18/05/2025	La versión es validada por control de calidad.	Sebastián Ríos	Tester
v 0.5	18/05/2025	19/05/2025	La versión es aprobada y continua la implementación.	José Cárdenas	Analista de Software
v 0.6	20/05/2025	22/05/2025	Se implementa el sistema con los requerimientos establecidos (HTML5, CSS, Python y MVC).	Silvia Contreras	Arquitecta de Software
v 0.6	23/05/2025	25/05/2025	La versión es validada por control de calidad.	Omar Moreno	Tester
v 0.6	25/05/2025	26/05/2025	La versión es aprobada y continua la implementación.	Hugo Ramírez	Analista de Software
v 1.0	27/05/2025	28/05/2025	Se lanza la versión definitiva con las características implementadas de acuerdo al contexto.	Silvia Contreras	Arquitecta de Software
v 1.0	28/05/2025	29/05/2025	Se realizan pruebas por control de calidad para detectar errores o defectos.	Mario Cervantes	Ingeniero en Calidad de Software
v 1.0	29/05/2025	30/05/2025	La versión es aprobada y se entrega al usuario final.	José Cárdenas	Analista de Software
v 1.0	30/05/2025	31/05/2025	El usuario final prueba el producto y confirma errores.	María Josefina	Arquitecta de Software
v 1.0	01/06/2025	02/06/2025	Se revisan los errores que el usuario final encontró y se corrigen.	Omar Moreno	Tester
v 1.1.1	02/06/2025	03/06/2025	Se realizan pruebas por control de calidad para detectar errores o defectos.	Mario Cervantes	Ingeniero en Calidad de Software
v 1.1.1	03/06/2025	04/06/2025	La versión es aprobada y se entrega al usuario final.	Silvia Contreras	Arquitecta de Software
v 1.1.1	05/06/2025	06/06/2025	La versión no presenta errores y queda a la espera de futuras actualizaciones.	María Josefina	Arquitecta de Software

Conclusión

Con todas las actividades realizadas nos damos cuenta de que es importante llevar un control casi milimétrico de lo que se hace, en estos casos, en el que se trabaja en un software con un equipo de personas nos da a entender que en muchas ocasiones llevar un orden puede resultar tedioso o muy complicado, para ello se toman diferentes medidas en lo que respecta los diferentes casos, al iniciar con un desarrollo de software se analizan las posibilidades, las ventajas y desventajas que puede tener, así como el presupuesto que se tiene, definir los problemas y darles soluciones, asignar los roles y gestionar los tiempos para que todo marche de acuerdo a lo que se espera y sobre todo documentar y guardar incidencias que puedan ocurrir en el proceso, recordar que estamos en un software y en ello tienen muchos problemas, nada de esto es perfecto y el más mínimo error puede ocasionar bastantes detalles que pueden afectar el proyecto final.

Link de GitHub

<https://github.com/UZLOP984/Aseguramiento-de-la-Calidad-de-Vida.git>

Referencias

Thales Group. (s. f.). *Software Versioning Basics | What is Software Versioning?* Thales Cloud Security Products. <https://cpl.thalesgroup.com/es/software-monetization/software-versioning-basics>

Cronograma: ¿qué es y cómo se crea? (2022, 5 julio). Personio.
<https://www.personio.es/glosario/cronograma/>