

Plan de verificación y validación de Software de “FemmeCode”

Introducción

El presente plan de verificación y validación es aplicable a los proyectos de software de “FemmeCode”

El contenido de este plan está basado en el estándar de verificación y validación para sistemas, software y hardware de la IEEE 1012-2016.

Propósito:

Este plan establece las actividades de verificación y validación de los requerimientos y el código de software desarrollados por la compañía "FemmeCode". Su propósito es ofrecer orientación sobre los procesos y herramientas de verificación y validación que se aplicarán en estos productos

Descripción general de las actividades de verificación y validación

El ejecutivo de SQA asumirá la responsabilidad de llevar a cabo las actividades de verificación y validación, garantizando el cumplimiento del proceso establecido y la integridad de los productos definidos en el plan. Su función principal será supervisar y asegurar que se sigan todas las etapas del plan, tanto durante su ejecución como después de ella, para asegurar la calidad y completitud de los productos resultantes.

La unidad de SQA estará bajo el mando del ejecutivo de SQA colaborando en el proyecto. El ejecutivo de SQA podrá distribuir actividades a otros colaboradores sin embargo él seguirá siendo el responsable de que dichas actividades se realicen de la manera correcta.

El ejecutivo de SQA deberá participar en la planificación del proyecto y en cada una de las iteraciones definidas dentro del ciclo de desarrollo del proyecto.

- **Aprobación del plan de verificación y validación**

Los responsables de aprobar el plan de verificación y validación serán el director de proyectos en conjunto con el cliente del proyecto.

- **Liberación del producto**

La liberación del producto para ser entregado al cliente estará a cargo del ejecutivo de SQA. Una vez que esta persona considere que todas las actividades y productos del plan han sido realizadas de manera satisfactoria enviará el visto bueno para la entrega del producto al cliente.

Tareas, métodos y criterios para la verificación y validación.

Tareas

- Revisión de requisitos: Se verifica que los requisitos del software estén claros, completos, consistentes y verificables.
- Diseño estático: Se realiza una revisión del diseño del software para identificar posibles problemas, como errores de diseño, ineficiencias o violaciones de estándares de codificación.
- Inspección de código: Se examina el código fuente para identificar errores de sintaxis, violaciones de estándares de codificación, mala estructura del código y posibles problemas de rendimiento.
- Pruebas unitarias: Se realizan pruebas a nivel de componente o módulo individual para verificar que cada parte del software funcione correctamente de manera aislada.
- Verificación de integración: Se prueban las interfaces y la interacción entre los diferentes componentes o módulos del software para asegurarse de que se integren correctamente.

Tareas de Validación:

- Pruebas funcionales: Se llevan a cabo pruebas para verificar que el software cumpla con las funcionalidades y los requisitos establecidos. Se diseñan casos de prueba que cubren diferentes escenarios y se comprueba si el software produce los resultados esperados.
- Pruebas de rendimiento: Se evalúa el rendimiento del software bajo diferentes cargas y condiciones para asegurarse de que cumple con los requisitos de rendimiento establecidos, como velocidad de respuesta, tiempo de procesamiento y uso eficiente de recursos.
- Pruebas de usabilidad: Se realizan pruebas para evaluar la facilidad de uso y la experiencia del usuario del software. Se recopilan comentarios y retroalimentación de los usuarios para identificar posibles mejoras en la usabilidad.
- Pruebas de seguridad: Se llevan a cabo pruebas para identificar vulnerabilidades y evaluar la seguridad del software. Esto puede incluir pruebas de penetración, pruebas de robustez, pruebas de autenticación y otras evaluaciones de seguridad.

Métodos comunes:

- Pruebas manuales: Los casos de prueba se ejecutan manualmente, y los resultados se verifican y registran de forma manual.
- Pruebas automatizadas: Se utilizan herramientas y scripts automatizados para ejecutar casos de prueba y verificar los resultados.

Esto agiliza el proceso de pruebas y facilita la repetición de las pruebas en diferentes versiones del software.

Criterios de Verificación y Validación:

- Cumplimiento de requisitos: Se verifica que el software cumpla con todos los requisitos establecidos, incluidos los funcionales, no funcionales y técnicos.
- Cumplimiento de estándares: Se verifica que el software cumpla con los estándares y las mejores prácticas de la industria, como los estándares de codificación, los estándares de seguridad y los estándares de usabilidad.
- Calidad del software: Se evalúa la calidad del software en términos de confiabilidad, rendimiento, usabilidad, seguridad y mantenibilidad.
- Ausencia de defectos: Se verifica que el software esté libre de defectos críticos y que los problemas identificados durante las pruebas se resuelvan de manera adecuada.

Alcance de las pruebas de validación

El alcance de las pruebas de validación en un plan de Verificación y Validación tiene como objetivo garantizar que el software cumpla con los requisitos de referencia establecidos. A continuación, se mencionan algunos aspectos clave que se deben considerar en el alcance de las pruebas de validación en los proyectos desarrollados por la empresa “FemmeCode”:

1. Requisitos de referencia: El primer paso es identificar y comprender los requisitos de referencia que se han establecido para el software. Estos requisitos pueden provenir de diversas fuentes, como el cliente, las normas o estándares de la industria, y las regulaciones aplicables. Es fundamental tener una comprensión clara y precisa de estos requisitos antes de diseñar y ejecutar las pruebas de validación.
2. Funcionalidad del software: Las pruebas de validación deben abordar la funcionalidad del software y asegurarse de que cumpla con los requisitos establecidos. Esto implica verificar que todas las características y funcionalidades especificadas en los requisitos de referencia estén presentes y funcionen correctamente. Se deben diseñar casos de prueba que cubran todos los escenarios relevantes para validar la funcionalidad del software.
3. Interoperabilidad: Si el software debe interactuar con otros sistemas, es importante realizar pruebas de interoperabilidad. Esto implica verificar que el software se integre adecuadamente con los sistemas y componentes externos, como bases de datos, servicios web u otros dispositivos. Las pruebas de interoperabilidad garantizan que la comunicación y el intercambio

de datos se realicen correctamente y que el software funcione de manera esperada en un entorno real.

4. Rendimiento y escalabilidad: Dependiendo de los requisitos de referencia, las pruebas de validación también pueden abordar el rendimiento y la escalabilidad del software. Esto implica evaluar cómo responde el software ante cargas de trabajo significativas, cómo maneja grandes volúmenes de datos y cómo se comporta en términos de velocidad y capacidad. Las pruebas de rendimiento y escalabilidad ayudan a garantizar que el software pueda satisfacer los requisitos de desempeño establecidos.
5. Usabilidad: La usabilidad es un aspecto crítico en la aceptación y adopción del software. Las pruebas de validación deben incluir evaluaciones de usabilidad para verificar que el software sea intuitivo, fácil de usar y satisfaga las necesidades y expectativas de los usuarios finales. Esto puede implicar pruebas de accesibilidad, pruebas de facilidad de navegación y pruebas de eficiencia en la realización de tareas comunes.
6. Seguridad y cumplimiento: En función de los requisitos de referencia, las pruebas de validación pueden abordar la seguridad y el cumplimiento normativo del software. Esto implica verificar que el software cumpla con los estándares y regulaciones de seguridad aplicables, protegiendo los datos y asegurando la confidencialidad, integridad y disponibilidad de la información. Se pueden realizar pruebas de penetración, pruebas de vulnerabilidad y otras evaluaciones de seguridad para validar la robustez del software.

Procesos de verificación y validación

Para la verificación del código y de que los requerimientos se hayan cumplido, se utilizarán las revisiones por pares, en donde se seleccionará a revisores que no hayan participado en la codificación del proyecto y que tengan suficiente experiencia para poder realizar la revisión.

Al terminar la revisión, los revisores deberán de llenar la siguiente “Matriz de verificación de requerimientos” en donde se registrar si el requerimiento fue o no cumplido como se especifico por el cliente y comentarios al respecto.

Matriz de verificación de requerimientos

Requerimiento	¿Fue cumplido como se especificó?	Comentarios
---------------	-----------------------------------	-------------

	(Si/No)	
RF001		
RF002		
RF003		
RF004		
RF005		

RF006		
RF007		
RF008		

RF009		
--------------	--	--

RF010		
RF011		
RF012		

RF013		
RF014		
RF015		
RF016		
RF017		
RF018		

RF019		
--------------	--	--

RF020		
RF021		

Etapas que requieren la revisión del cliente:

El objetivo de estas revisiones es obtener la retroalimentación y la aprobación del cliente, y asegurar que el software esté alineado con sus necesidades y expectativas:

1. **Revisión de requisitos:** Antes de comenzar el desarrollo del software, es fundamental realizar una revisión de requisitos con el cliente. En esta etapa, se verifica que los requisitos del cliente estén claros, comprensibles y alineados con las expectativas. El alcance de verificación incluiría asegurarse de que los requisitos estén documentados correctamente, sean verificables y no presenten contradicciones o ambigüedades.
2. **Diseño preliminar:** Después de establecer los requisitos, se realiza una revisión del diseño preliminar con el cliente. En esta etapa, se presentan los conceptos y la arquitectura del software. El alcance de verificación involucraría revisar que el diseño cumpla con los requisitos establecidos, que sea factible técnicamente y que se hayan considerado las restricciones y las necesidades del cliente.
3. **Diseño detallado:** Una vez aprobado el diseño preliminar, se procede a la etapa de diseño detallado. En esta fase, se elaboran especificaciones técnicas más precisas y se definen los componentes, las interfaces y los

algoritmos en detalle. El alcance de verificación incluiría revisar que el diseño detallado se ajuste al diseño preliminar aprobado y que cumpla con los requisitos y estándares establecidos.

4. Implementación del código: Después de finalizar el diseño detallado, se inicia la implementación del código. Antes de la revisión del cliente, se debe realizar una verificación exhaustiva del código para garantizar su calidad y cumplimiento de los estándares. Esto puede incluir la revisión de la sintaxis, las convenciones de codificación, la modularidad y la legibilidad del código.
5. Pruebas unitarias: Las pruebas unitarias se llevan a cabo para verificar que cada componente o módulo individual funcione correctamente según lo especificado. Antes de la revisión del cliente, todas las pruebas unitarias deben completarse y los resultados deben ser satisfactorios. El alcance de verificación aquí implica asegurarse de que todas las funcionalidades y las rutas del código se hayan probado adecuadamente.

Requisitos de documentación de pruebas:

Los requisitos de documentación de las pruebas de verificación y validación deben incluir los planes de prueba, los diseños, los casos, los procedimientos y los resultados para las pruebas de componentes, integración, calificación y aceptación desarrolladas por el esfuerzo de verificación y validación. La documentación de la prueba de verificación y validación debe cumplir con el propósito, el formato y el contenido del documento de prueba definido por el proyecto (p. ej., IEEE Std 829-2008 [B3]). Si el esfuerzo de verificación y validación utiliza documentación de prueba o tipos de prueba diferentes a los de este estándar (es decir, componente, integración, calificación y aceptación), el esfuerzo de V&V deberá mostrar un mapeo de la documentación de prueba propuesta y la ejecución de los elementos de prueba definidos en este estándar.