




LABORATORIO

PRUEBAS de Software



FAVA - Formación en Ambientes Virtuales de Aprendizaje

SENA - Servicio Nacional de Aprendizaje.

Estructura de contenidos

Pág.

Introducción	3
Objetivos	4
Desarrollo de contenidos	4
Consideraciones	5
Procedimiento	6
Pasos	7
Ejercicios	8
Evidencias	9
Glosario	10
Bibliografía	11
Control del documento	12

Laboratorio: pruebas de software

Introducción



El uso de buenas prácticas en el desarrollo de software, requiere de un proceso de verificación y validación (V&V) adecuado para que el producto o servicio final alcance el nivel de calidad esperado por el cliente. En V&V, las pruebas de software o testing son fundamentales, pues proveen al desarrollador de un conjunto de técnicas dinámicas que permiten la identificación y depuración de fallos en el sistema construido.

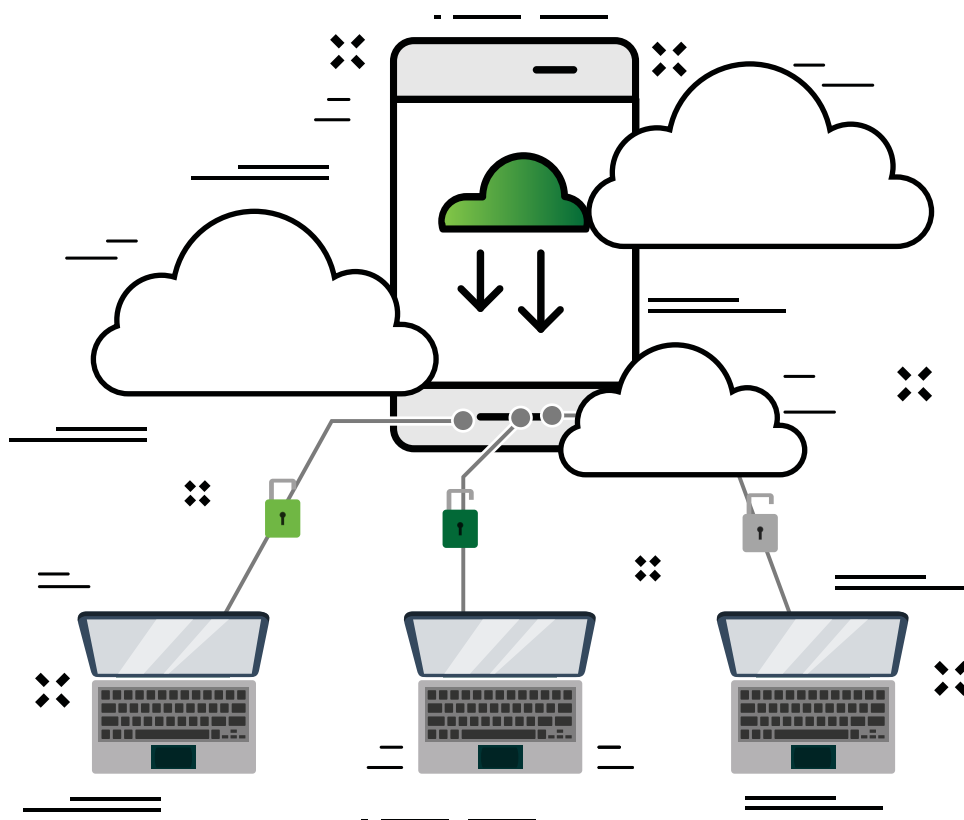
Es necesario que, como desarrollador de aplicaciones de software, en su proceso de formación académica adquiera las competencias para el diseño adecuado de un plan de pruebas, definiendo los casos de prueba y sus requerimientos.

En el presente laboratorio se presenta una situación problemática de aprendizaje, con actividades prácticas, mediante las cuales se busca el afianzamiento de los fundamentos teóricos relacionados con las pruebas de software.

Desarrollo de contenidos

Objetivos

- Desarrollar actividades prácticas, que permitan el afianzamiento de conceptos relacionados con las pruebas de software.
- Aplicar los fundamentos teóricos presentados en el material de apoyo, para el desarrollo de actividades que ayuden al entrenamiento en el diseño y aplicación de casos de prueba.
- Implementar casos de prueba, que permitan identificar elementos fundamentales en el diseño del plan de pruebas de software.
- Utilizar técnicas tradicionales en pruebas de software, basadas en la ejecución del código con el enfoque de caja blanca, fundamentales en la identificación de errores lógicos.



Consideraciones

Para el desarrollo del laboratorio es importante que se tenga en cuenta los siguientes aspectos relacionados en la tabla:

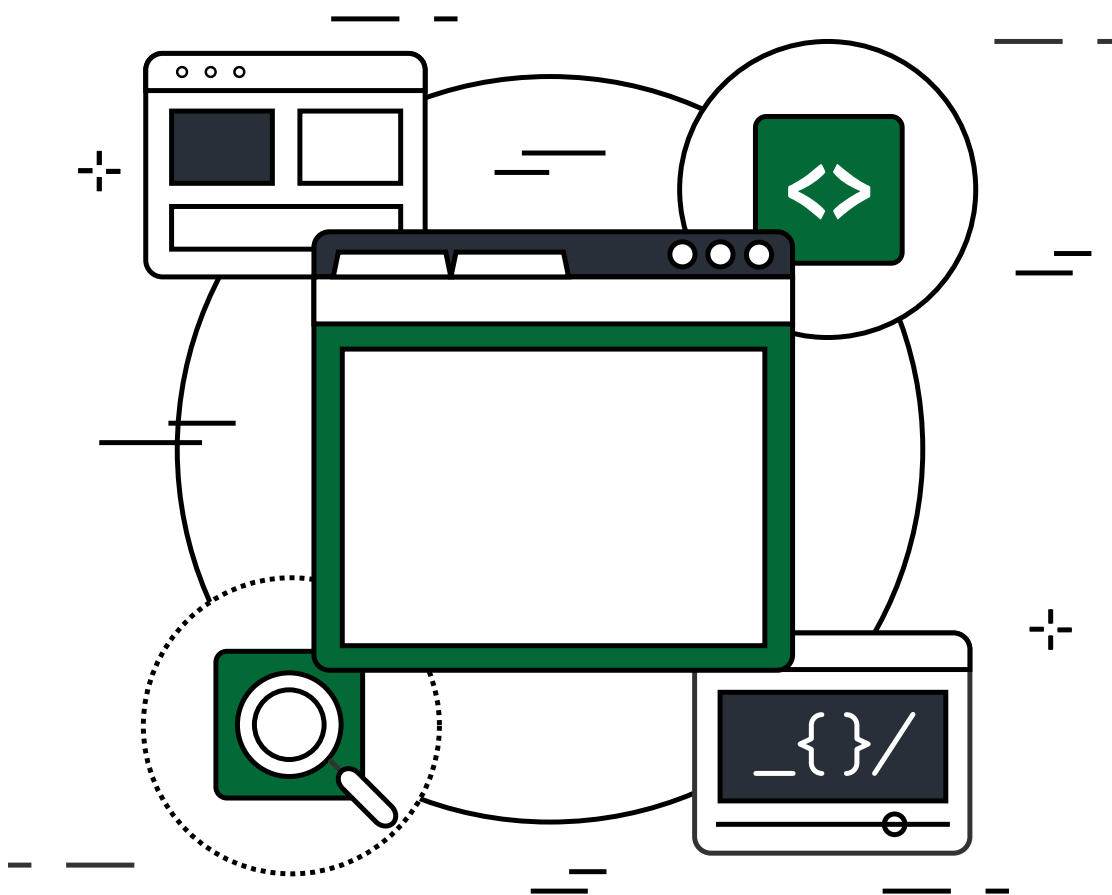
ÍTEM	DESCRIPCIÓN
Soporte Teórico	Objeto de aprendizaje Pruebas de software.
Productos requeridos	Enunciados de los ejercicios a desarrollar. Plantilla de Caso de prueba.
Herramienta SW	Ambiente de desarrollo seleccionado y herramienta ofimática de Excel.

Procedimiento

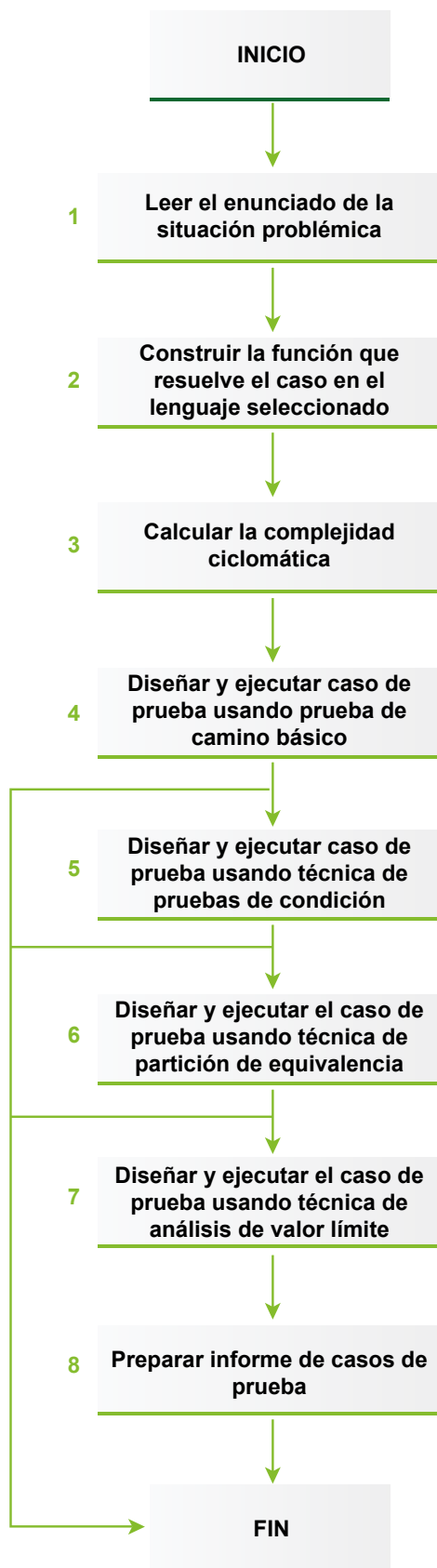
Las siguientes actividades propuestas en el laboratorio, son de tipo práctico que buscan el entrenamiento del aprendiz en el diseño de casos de prueba para la evaluación de la lógica interna de un programa.

Se aplicarán técnicas de caja blanca en forma tradicional, que permitirán la construcción de la base de conocimiento necesaria para la implementación de buenas prácticas en las pruebas de software.

Los aprendizajes darán al estudiante lineamientos para incorporar en el proceso de pruebas si lo prefiere, software especializado que facilitará y ampliará las opciones en el diseño de pruebas.



Pasos



Ejercicios:

1. Analizar la siguiente situación problemática: una empresa que fabrica piezas torneadas para vehículos, paga a sus empleados de acuerdo a las horas trabajadas y a una tarifa de pago por hora, de acuerdo a la siguiente tabla:

Categoría	Tarifa
1	12000
2	17000
3	22000

Si la cantidad de horas trabajadas es mayor a 40 horas, la tarifa se incrementa en un 25% para las horas extras. Calcular el salario del trabajador dadas las horas trabajadas y la categoría.

2. Construir una función que resuelva el caso presentado en la situación problemática e implementarla en el lenguaje de programación deseado.
3. Calcular la complejidad ciclomática, determinar los caminos resultantes, y casos de prueba según la técnica de prueba del camino básico.
4. Diseñar casos de prueba usando la técnica de pruebas de condición.
5. Presentar informe de los casos de prueba, usando el instrumento Plantilla_Caso_de_prueba.xls.

Evidencias

- a. Código del programa que implemente la función solicitada.
- b. Documento con:
 - Cálculo de la complejidad ciclomática.
 - Caminos resultantes según la técnica del camino básico. Construir grafo de flujo.
 - Casos de prueba según la técnica de pruebas de condición.
 - Casos de prueba según la técnica de partición de equivalencia.
 - Casos de prueba según la técnica de análisis de valores límite.
- c. Plantilla con el registro de los casos de prueba.

NOTA: el laboratorio no es calificable hace parte de una actividad complementaria necesaria para construir la evidencia de aprendizaje, lo cual implica que para desarrollar la evidencia es necesario apropiarse del conocimiento de este laboratorio.



Glosario

Función: microprograma o módulo dentro de un programa, que contiene bajo un solo nombre pequeñas rutinas que pueden ser utilizadas una o varias veces.

Grafo: representación gráfica que representa relaciones binarias entre elementos de un conjunto, por medio de nodos y enlaces.

Complejidad ciclomática: medida que permite determinar el número de caminos a probar usando la técnica de camino básico.

Bibliografía

BOLAÑOS, D., SIERRA, A., & ALARCÓN, M. (2008). *Pruebas de Software y JUnit*. Madrid: Pearson Prentice Hall.

PRESSMAN, R. (2006). *Ingeniería del Software: Un enfoque práctico*. Sexta edición. McGrawHill.

IEEE Standard Glossary of Software Engineering Terminolgy [IEEE, 1990].

Control del documento

CONSTRUCCIÓN
OBJETO DE
APRENDIZAJE



LABORATORIO PRUEBAS DE SOFTWARE

Centro Industrial de Mantenimiento Integral - CIMI
Regional Santander

Líder línea de producción: Santiago Lozada Garcés

Asesores pedagógicos: Rosa Elvia Quintero Guasca - V2
Claudia Milena Hernández Naranjo - V2

Líder expertos temáticos: Rita Rubiela Rincón Badillo

Experto temático: José Ricardo Arismendi Santos - V1
Experto temático: Edward Jose Beltran Lozano - V2

Diseño multimedia: Jesús Antonio Vecino Valero

Programador: Francisco José Lizcano Reyes

Producción de audio: Victor Hugo Tabares Carreño



Este material puede ser distribuido, copiado y exhibido por terceros si se muestra en los créditos. No se puede obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de la licencia que el trabajo original.