

Facultad de Ingeniería y Tecnología Informática
Licenciatura en Sistemas de Información
Materia: Testeo y Prueba de Software
Año 2025

Trabajo Práctico Final
Elaboración y ejecución de un plan de pruebas

Objetivo: Elaborar y ejecutar un plan de pruebas de código y funcionales sobre un software desarrollado o propuesto.

El objetivo principal es afianzar los conceptos básicos de la materia mediante una prueba real, con software reales

Consigna:

- Elegir una aplicación desarrollada en python o java, con el objetivo de realizar pruebas de código (unitarias, caja blanca) y pruebas de aceptación.
- El lenguaje es importante porque permite que la aplicación sea probada utilizando herramientas de testing automatizado como pytest o junit.
- Es requerido que tenga una interfaz gráfica que permite también realizar pruebas de aceptación automatizadas (ej. selenium o katalon).
- En caso de no poder realizar pruebas de aceptación automatizadas (quizas porque la interfaz no lo permite, por ej. si se usó Flask), las mismas deben llevarse a cabo manualmente.
- Las pruebas de aceptación son pruebas "de usuario final" para encontrar errores funcionales desde la interfaz.
- Las pruebas de código son para demostrar que las funciones internas hacen lo que tienen que hacer, independientemente de la funcionalidad principal. Recomendación: se debe buscar una función simple de probar.
- Se debe entregar el trabajo final en un pdf, con toda la información de soporte, incluyendo el plan de pruebas ejecutado (excel) y toda la evidencia de ejecución (printscreens, por ej.).
- Para la elaboración del plan de pruebas se debe investigar en internet diferentes modelos existentes, tratando que la información permita demostrar las funciones probadas, los casos de prueba, los lotes utilizados, los pasos realizados en cada caso, las precondiciones, el resultado esperado y el real y la tipificación de la prueba que no resultó.
- Para tipificar el resultado erróneo de la prueba (algo que el sistema debía hacer y no hizo), se debe definir un criterio de evaluación. Se propone algo así como: bloqueante (no permite continuar la ejecución), crítico (pérdida de información), mayor (menor criticidad sin perdida de información), menor (bajo impacto sin perdida), trivial (solo estético).

Estructura del trabajo final

Indice

1. Objetivo (trabajo)
2. Alcance (trabajo)
3. Desarrollo..
 - Descripción de la aplicación a probar.
 - Objetivo y alcance de las pruebas.
 - Especificación de las pruebas.
 - Plan de Pruebas.
 - Ejecución de las pruebas. Descripción de cómo se elaboró el plan de ejecución. Incluir los pasos a seguir para ejecutar la aplicación (condiciones como por ejemplo, la versión de python utilizada).
 - Conclusión (de la ejecución de las pruebas)
4. Anexos
 - Plan de ejecución de pruebas XXX. Version X (excel).
 - evidencia-ejecución (paquete zip).

Ejemplo de como nombrar la evidencia de los casos de prueba ejecutados.

Caso #1 - Alta de libro - prueba ok

Caso #2 - Alta de libro - isbn invalido.

Entregables:

1. Trabajo práctico con el detalle de la planificación y ejecución de las pruebas, con toda la información, basado en el índice propuesto anteriormente.
2. Plan de pruebas.
3. Evidencia de la ejecución de las pruebas (resultado).
4. Evidencia de soporte a la ejecución de las pruebas.

Criterios de evaluación:

1. Completitud en las consignas.
2. Claridad en la redacción.
3. Fundamentación del contenido, enfocado en las metodologías y técnicas utilizadas para la elaboración del trabajo final.