



Nombre del proyecto: Parkimaniacos



Protocolo del Proceso de Testing

Equipo de trabajo:

José Ángel García Arce	NC: 17130786
Luis Emmanuel Méndez Barrios	NC: 17130804
Enrique Antonio Belmarez Meraz	NC: 17130765
José Baltazar Martínez De La Rosa	NC: 17130049

Introducción:

El protocolo del proceso de testing es el encargado de gestionar todo lo relacionado con el proceso de prueba del proyecto. Las pruebas son básicamente un conjunto de actividades dentro del desarrollo de software. Dependiendo del tipo de pruebas, estas actividades podrán ser implementadas en cualquier momento de dicho proceso de desarrollo. Existen distintos modelos de desarrollo de software, así como modelos de pruebas. A cada uno corresponde un nivel distinto de involucramiento en las actividades de desarrollo.

En el desarrollo de software existen dos diferentes tipos de pruebas las cuales son las siguientes:

Pruebas estáticas

Son el tipo de pruebas que se realizan sin ejecutar el código de la aplicación.

Puede referirse a la revisión de documentos, ya que no se hace una ejecución de código. Esto se debe a que se pueden realizar "pruebas de escritorio" con el objetivo de seguir los flujos de la aplicación.

Pruebas dinámicas

Todas aquellas pruebas que para su ejecución requieren la ejecución de la aplicación.

Las pruebas dinámicas permiten el uso de técnicas de caja negra y caja blanca con mayor amplitud. Debido a la naturaleza dinámica de la ejecución de pruebas es posible medir con mayor precisión el comportamiento de la aplicación desarrollada.

Estos tipos de pruebas expuestos anteriormente son los que se estarán haciendo a el proyecto con el fin de garantizar el correcto funcionamiento del mismo a continuación se estará expresando como se realizaran cada una de estas.

Prueba estática:

Como se menciona en la página anterior para este tipo de pruebas lo primordial no es ejecutar el código sino la revisión de los diferentes documentos que componen el proyecto, así como también los patrones de diseño seguidos durante la creación del código y obviamente valga la redundancia la creación del código.

Por lo cual la implementación de dichas pruebas es fundamental a la hora de asegurarnos de la calidad de el producto en cuestión, por lo cual en este caso las pruebas serán sobre este tipo de documentos, así como del diseño de la aplicación siguiendo los estándares y las normas necesarias para seguir con los lineamientos previamente establecidos por el cliente que a su vez satisfacen nuestros estándares de calidad.

Caja Negra:

Las pruebas de caja negra, es una técnica de pruebas de software en la cual la funcionalidad se verifica sin tomar en cuenta la estructura interna de código, detalles de implementación o escenarios de ejecución internos en el software.

En las pruebas de caja negra, nos enfocamos solamente en las entradas y salidas del sistema, sin preocuparnos en tener conocimiento de la estructura interna del programa de software. Para obtener el detalle de cuáles deben ser esas entradas y salidas, nos basamos únicamente en los requerimientos de software y **especificaciones funcionales**.

Ejemplo de plantilla para caso de Prueba de Caja Negra:

ID	Caso de Prueba	Descripción	Fecha	Área Funcional	Funcionalidad
Aplicación Movil	Cronometro	Verificar la Funcionalidad del Cronometro	11/10/2020	Desarrollo	Funciona a la perfección ya que si corre el tiempo
Aplicación web	Base de datos	Verificar las estadísticas de uso de los parquímetros	12/10/2020	Desarrollo	Funciona ya que nos muestra en términos numéricos la cantidad de personas dependiendo la hora

Aplicación Movil	Métodos de pago	Verificar los métodos de pago	11/10/2020	Análisis	Funciona ya que nos permite pagar

Caja Blanca:

En programación, se denomina **cajas blancas** a un tipo de pruebas de software que se realiza sobre las funciones internas de un módulo. Así como las pruebas de caja negra ejercitan los requisitos funcionales desde el exterior del módulo, las de caja blanca están dirigidas a las funciones internas. Entre las técnicas usadas se encuentran; la cobertura de caminos (pruebas que hagan que se recorran todos los posibles caminos de ejecución), pruebas sobre las expresiones lógico-aritméticas, pruebas de camino de datos (definición-uso de variables), comprobación de bucles (se verifican los bucles para 0,1 e interacciones, y luego para las interacciones máximas, máximas menos uno y más uno).

Ejemplo de plantilla para realizar pruebas de caja blanca:

```
CREATE TABLE FormaPagos(
id_Pago int PRIMARY KEY NOT NULL IDENTITY(1,1) ,
Nombre_Titular varchar(30) NOT NULL,
Numero_Tarjeta int NOT NULL,
Codigo_Seguridad int NOT NULL,
Fecha_ExpiracionM SMALLINT NOT NULL,
Fecha_ExpiracionA SMALLINT NOT NULL,
id_cliente int FOREIGN KEY REFERENCES Clientes(id_cliente)
)
```

DESCRIPCION PRUEBA CAJA BLANCA				
DATOS DE ENTRADA			RESULTADO	
Query para crear la table Forma de Pago			tabla Forma de Pago	
Tipo de flujo de datos				
Archivo	×	Pantalla	Informe	Formulario
La estructura de datos que viaja con el flujo				
Query Forma de Pago				
COMENTARIOS:				
El código de la Query cumple con el estándar establecido.				