



Plan de pruebas

Para el trabajo de grado Ane-Stent

Stephanie Domínguez Andrade

s.dominguez@javeriana.edu.co

Juan Sebastián Espinosa Torres

espinosa_j@javeriana.edu.co

Jose Antonio Quintero Gómez

j.quinterog@javeriana.edu.co

David Alonso Villamizar Lizcano

villamizar.david@javeriana.edu.co

1. Historial de cambios

Versión	Descripción	Fecha	Responsable
1.0	Versión inicial del documento	03/02/2018	Stephanie Dominguez

2. Prefacio

Este documento pretende definir los conceptos que el equipo de ANE-STENT utilizara para llevar a cabo las pruebas en el prototipo asegurando la calidad de software. Las plantillas descritas en este documento son para las pruebas unitarias, funcionales y de integración.

3. Tabla de contenidos

1. Historial de cambios.....	1
2. Prefacio.....	1
3. Tabla de contenidos.....	1
4. Lista de tablas.....	2
6. Alcance	2
7. Definiciones	2
8. Pruebas del sistema	2
9. Documentación de las pruebas	2
9.1. Pruebas de rendimiento	4
10. Métricas de medición	5
11. Reporte pruebas	5
12. Referencias	6

4. Lista de tablas

Tabla 1. Tabla de definiciones	2
Tabla 2. Características	4
Tabla 3. Características prueba de rendimiento	5

6. Alcance

El documento describe las pruebas que se van a realizar, esta parte incluye el ambiente de hardware y de software en el cual opera el proyecto. También se incluye la estructura de la documentación de la ejecución de las pruebas y el reporte de las pruebas.

7. Definiciones

Termino	Descripción
Calificación de la prueba	Las pruebas se deben calificar de la siguiente manera: pendiente, aprobada y fallo
Personas responsables	Persona encargada de las hacer las pruebas asignada.
Tiempo	Tiempo en minutos de la duración de la prueba
Tipo de prueba	Pruebas unitarias, funcionales, rendimiento, subjetiva y de integración
Valores limites	Dados unos valores, el sistema tiene que validar que el programa los acepte o rechace, por ejemplo, valores negativos.
Outputs	El resultado de una prueba.

Tabla 1. Tabla de definiciones

8. Pruebas del sistema

En esta sección, se describe qué manera, detallada qué tipo de pruebas se realizará, quien la ejecuta, los ambientes de software y hardware, las técnicas a utilizar y cuando se realizan [(Sandeep Desai, 2016)]. (Ver anexo descripción de pruebas del sistema)

9. Documentación de las pruebas

En esta sección se encuentra descrita de manera detallada las métricas que se usarán, para evaluar el código (IEEE, 2014). (Ver anexo plantilla de pruebas de sistema)

La siguiente tabla muestra las características de las plantillas de pruebas unitarias, funcionales, de rendimiento y de integración:

Nombre	Descripción
Id del caso de la prueba	Se va identificar cada caso de prueba realizado con id únicos, por ejemplo las pruebas unitaria usarán: PU-001, para la pruebas funcionales: PF-001, prueba de integración: PI-001, prueba de rendimiento PR-001
Persona responsable	Este campo, indica la persona que está encargada de diseñar y hacer la prueba.
Descripción de la prueba	Da al lector una visión general de la prueba.
Clase	Este campo es para las pruebas unitarias, indica la clase que se está probando.
Metodo	Este campo es para las pruebas unitarias, indica el método que se está probando.
Pasos de las pruebas	Indicar los pasos que realizo la persona para completar la prueba.
Datos de entrada	Los datos utilizados para probar el método/funcionalidad/integración del sistema.
Valor de salida esperado	Este valor es el deseado al final de la prueba.
Valor de salida obtenido	Este valor es el real al final de la prueba.
Errores encontrados	Indica cuántos errores se han encontrado en el código.

Errores corregidos	Si se encontraron errores, en este campo se indica si se corrigió, en dado caso se cambia su estado de fallo a aprobado
Comentario	Si la persona responsable tiene una sugerencia para el código, deberá usar este campo
Estado	Para las pruebas, se usarán varias convenciones para definir este campo: Aprobado: No se encontraron errores. Fallo: Hay errores en el funcionamiento del código. Pendiente: No se ha realizado la prueba.
Requerimiento	Este campo es para las pruebas funcionales, indica el requerimiento asociado.
Módulos asociados	En las pruebas de integración y funcionales se colocan los módulos que está siendo evaluado
Métrica	Este campo indica la cantidad de errores, y le da un valor: Más de tres errores: Pésima calidad. Entre 1-3 errores: mala calidad. 0: excelente calidad.

Tabla 2.Características

9.1. Pruebas de rendimiento

Las pruebas de rendimiento tienen un carácter especial ya se debe medir el tiempo con los cuerpos suaves y rígidos, que no se divide en plantillas como las anteriores, se dividen en diferentes hojas donde se mencionan:

Nombre	Descripción
--------	-------------

Id de la hoja	Se va a identificar cada caso de prueba PR-001, más el objetivo de la prueba, por ejemplo, PR-001, Promedio rígido
Id del archivo	Este será el identificador que tiene el archivo, que se identifica por cada hoja, por ejemplo, AVERAGE of bunny.vtk
Tiempo del archivo	Este campo equivale a todos los tiempos que debe sacar el sistema. Para luego compararlo
Tiempo archivo único	Este campo equivale a los tiempos máximos, mínimos y promedio de cada archivo

Tabla 3. Características prueba de rendimiento

10. Métricas de medición

Para medir la calidad de código, se utilizará como métricas, el número de errores cometidos durante las pruebas, esto tendrá como consecuencia conocer si el componente es de buena calidad. Los intervalos son los siguientes:

- Más de tres errores: Pésima calidad.
- Entre 1-3 errores: mala calidad.
- 0: excelente calidad.

Para las pruebas rendimiento, se definió un criterio de factibilidad para evaluar la prueba de concepto, este criterio es:

“la prueba de concepto en su última versión disponible del 20 de mayo de 2018 debe tener tiempo promedio de duración por cuadro en segundos inferior a 0.033, para modelos de hasta 100 vértices”.

Esta medida será usada para el reporte de pruebas.

11. Reporte pruebas

En esta sección se encuentra el análisis de los resultados obtenidos en las pruebas.

(Ver reporte de pruebas)

12. Referencias

- IEEE. (2014, 12 2). *IEEE Test plan template* . Retrieved from IEEE Test plan template :
<http://www.fit.vutbr.cz/study/courses/ITS/public/ieee829.html>
- Sandeep Desai, A. S. (2016). *SOFTWARE TESTING: A Practical Approach*. Delhi, India: PHI Learning.