Especificación de requerimientos de software

Para el trabajo de grado Ane-Stent

Stephanie Domínguez Andrade

[s.dominguez@javeriana.edu.co](mailto:s.dominguez@javeriana.edu.co)

Juan Sebastián Espinosa Torres

[espinosa\_j@javeriana.edu.co](mailto:espinosa_j@javeriana.edu.co)

Jose Antonio Quintero Gómez

[j.quinterog@javeriana.edu.co](mailto:j.quinterog@javeriana.edu.co)

David Alonso Villamizar Lizcano

[villamizar.david@javeriana.edu.co](mailto:villamizar.david@javeriana.edu.co)

J

# Historial de cambios

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Descripción** | **Fecha** | **Responsable** |
| 1.0 | Desarrollo inicial del documento | 13/11/2017 | Todos |
| 1.0 | Versión inicial del documento | 13/11/2017 | Juan Sebastián Espinosa |

# Prefacio

Este documento corresponde con la especificación de requerimientos de software (SRS) del prototipo a desarrollar en el trabajo de grado Ane-Stent de la modalidad de investigación.

El presente documento expone los requerimientos que deberá cumplir el prototipo que genera la simulación desarrollada al aplicar la tecnología que se explora, es necesario aclarar que el prototipo desarrollado en este trabajo de grado no es el producto final, al ser un proyecto de investigación, el resultado que se pretender obtener de este proceso de desarrollo es un informe de factibilidad de la aplicación de esta tecnología.

Para más información ver documento de propuesta de trabajo de grado.

# Tabla de contenidos

[1. Historial de cambios 1](#_Toc498374907)

[2. Prefacio 2](#_Toc498374908)

[3. Tabla de contenidos 3](#_Toc498374909)

[4. Lista de tablas 5](#_Toc498374910)

[5. Introducción 6](#_Toc498374911)

[5.1. Definiciones, acrónimos, y abreviaciones 6](#_Toc498374912)

[5.2. Apreciación global 6](#_Toc498374913)

[6. Descripción global 7](#_Toc498374914)

[6.1. Características del usuario 7](#_Toc498374915)

[6.1.1. Estudiantes 7](#_Toc498374916)

[6.1.2. Investigadores 7](#_Toc498374917)

[6.2. Restricciones 7](#_Toc498374918)

[6.3. Suposiciones y dependencias 9](#_Toc498374919)

[7. Requerimientos específicos 10](#_Toc498374920)

[7.1. Requerimientos funcionales 10](#_Toc498374921)

[7.2. Requerimientos no funcionales 10](#_Toc498374922)

[7.3. Priorización 10](#_Toc498374923)

[8. Proceso de ingeniería de requerimientos 10](#_Toc498374924)

[8.1. Mecanismos de priorización 10](#_Toc498374925)

[8.2. Procesos de construcción de requerimientos 11](#_Toc498374926)

[Obtención 12](#_Toc498374927)

[Verificación 12](#_Toc498374928)

[Análisis 12](#_Toc498374929)

[Especificación 12](#_Toc498374930)

[9. Proceso verificación 13](#_Toc498374931)

[10. Referencias 15](#_Toc498374932)

# Lista de tablas

[Tabla 1. Apreciación global. Indica la función de cada sección. 7](#_Toc498374933)

[Tabla 2. Restricciones. 8](#_Toc498374934)

[Tabla 3. Restricciones de diseño. 9](#_Toc498374935)

[Tabla 4. Fases de construcción de requerimientos. 11](#_Toc498374936)

[Tabla 5. Análisis requerimientos. 12](#_Toc498374937)

[Tabla 6. Campos de especificación de requerimientos. 13](#_Toc498374938)

[Tabla 7. Características de un buen requerimiento. 14](#_Toc498374939)

# Introducción

## Definiciones, acrónimos, y abreviaciones

Algunos términos presentes en este documento pueden no tener suficiente contexto para entenderse por sí mismos, por eso, se pueden consultar en el glosario.

(Anexo 1 Glosario)

## Apreciación global

Este documento tiene como fin mostrar a los integrantes del grupo que necesita tener el software que se tiene en mente realizar, al igual que como se debe realizar. Para los lectores externos, se informa que debe cumplir el software. En la tabla 1 se puede apreciar más detalladamente cada sección y su propósito.

|  |  |  |
| --- | --- | --- |
| Sección | Descripción | ¿Cuándo se va a usar? |
| Introducción | Esta parte se dedica a definir al lector una definición de los términos y acrónimos usados en el resto del documento. | Antes de leer el resto del documento, ya que algunos términos pueden tener varios significados y es importante que el lector entienda lo mismo que se quiso decir a la hora de escribir. |
| Descripción Global | Esta sección, indica las condiciones preliminares para el uso del producto. Esto es, para quienes va dirigido, qué restricciones se tienen y supuestos y dependencias que se hicieron para que el producto funcione. | Al momento querer hacer uso o preparar un escenario para usar el programa se debe tener en cuenta esta sección, ya que de no cumplirse lo descrito en esta sección, es probable que no funcione o no del modo correcto. |
| Requerimientos Específicos | Esta sección indica los requerimientos del software a realizar, los lista y define criterios de aceptación para los mismos. | En caso de los integrantes del grupo es necesario para ver como se debe desarrollar y en caso de los interesados en el software, para ver si cumple con lo esperado. |
| Proceso de verificación | Esta sección mostrará el proceso que se llevará a cabo para comprobar que un requerimiento planteado esté establecido correctamente. | Se usará principalmente por los integrantes del grupo para llevar control de los requerimientos. |

Tabla 1. Apreciación global. Indica la función de cada sección.

# Descripción global

## Características del usuario

Como este trabajo de grado busca ser una base para apoyar investigaciones futuras, los tipos de usuario que se espera son:

* **Estudiantes** a menos de dos semestres de graduarse que planean o desarrollan su trabajo de grado en el campo del grupo de investigación “BASPI”.
* **Investigadores** que planean o desarrollan investigaciones en el campo del grupo de investigación “BASPI”.

Como se espera que el producto de este trabajo de grado sea usado en una sola máquina y por un solo usuario a la vez, se establece que todos los usuarios tienen los mismos permisos de acceso a las funcionalidades e información que maneja el producto y cumplen los mismos roles al usarlo. También se espera que la frecuencia de uso para ambos tipos de usuario sea igual.

### Estudiantes

Se espera que tengan experiencia en el lenguaje C++, compilación usando CMake y Make y acceso a un computador con las especificaciones mencionadas en la sección de Restricciones de este documento. ([Ver sección 7.2.)](#_Restricciones)

### Investigadores

Se espera que tengan mínimo la misma experiencia de los usuarios estudiantes. Además, se espera que sean expertos en el área de bioingeniería y tengan conocimiento básico de cómo funcionan las simulaciones físicas.

## Restricciones

Existen tres tipos de restricciones: las generales, las de software y las de hardware. En la tabla 2 se especifican cada una para el proyecto.

|  |  |
| --- | --- |
| Restricciones generales | * El sistema debe desarrollarse en español * El sistema tiene una arquitectura *single-node*. * El programa debe estar implementado en el paradigma orientado a objetos. * El sistema debe ser standalone |
| Restricciones de software | * El prototipo debe desarrollarse con C++. * El prototipo debe realizarse con Bullet Physics. * El programa debe poder desplegarse en los sistemas operativos OSX Sierra, Ubuntu 14.04 LTS y Windows 10 |
| Restricciones de hardware | * El programa debe poder simular con 8 GB de RAM. * El prototipo debe poder simularse en un computador con memoria dedicada a gráficos 2 GB * El prototipo debe simular con frecuencia de procesador 2,00 GHz. * El prototipo debe simular en una pantalla con resolución de 1366x768 * Debe simular en una arquitectura de 64 Bits. |

Tabla 2. Restricciones.

En la tabla 3, se muestra las restricciones de diseño.

|  |  |
| --- | --- |
| Confiabilidad | Es prioridad que los usuarios confíen en la simulación |
| Tolerancia a fallos | En caso de que la simulación falle, esta debe volver a un estado seguro o reanudar el proceso |
| Concurrencia | Ya que se debe lograr un buen rendimiento en los prototipos, será necesario utilizar programación con hilos como métodos de concurrencia de procesos para lograrlo. (TheServerSide, 2013) |
| Lenguajes | Dado que el producto va a estar implementado usando BulletPhysics, se usará el lenguaje C++. |
| Interfaces externas | Se utilizará BulletPhysics y un motor para el renderizado (Plan de proyecto sección 8.1) |
| Tecnologías y herramientas | Al ser un proyecto no financiado, en su mayoría, se dependerá de software libre y software con licencias provistas por la Pontificia Universidad Javeriana. |
| Estándares de programación | Como estándares de código, se utilizará la guía de estilos de código para C++ de Google (Google LLC, 2017) como se estableció en el documento de plan de proyecto. (sección 11.2.2.2. De plan de proyecto). |

Tabla 3. Restricciones de diseño.

## Suposiciones y dependencias

Las siguientes son las suposiciones que pueden afectar los requerimientos de este documento:

* Se suponen todas las restricciones de la sección restricciones. ([Ver sección 7.2.](#_Restricciones))
* El equipo del trabajo de grado tendrá disponibilidad de 12 horas semanales desde el inicio hasta la entrega definitiva del trabajo de grado.
* El equipo de trabajo de grado tiene las capacidades y experiencia necesarias para cumplir los objetivos del trabajo de grado en el tiempo establecido.
* Es posible que no se logre realizar el producto que cumpla los requerimientos que se plantean en este documento.
* La interacción del usuario con el producto no necesariamente ocurrirá en tiempo real.
* Los usuarios del producto cumplirán tendrán las características descritas en la sección características del usuario. ([Ver sección 7.1.](#_Características_del_usuario))
* El producto se ejecutará en una sola máquina.
* El producto se usará por un solo usuario a la vez.

Las siguientes son dependencias del producto:

* Ubuntu 16 LTS.
* C++ con sus librerías por defecto.
* El compilador GCC.
* Bullet Physics.
* El motor de renderizado que se define en la siguiente fase.

# Requerimientos específicos

Con el fin de llevar a cabo la especificación de requerimientos, éstos debieron ser sometidos al proceso de ingeniería de requerimientos ([sección 9.](#_Proceso_de_ingeniería)).

## Requerimientos funcionales

Los requerimientos funcionales del producto se encuentran en la Tabla 1. (Ver la hoja “Funcionales” en el libro anexo 2 Especificación de Requerimientos).

Los requerimientos funcionales son aquellos que especifican qué funcionalidades provee el sistema.

## Requerimientos no funcionales

Los requerimientos funcionales del producto se encuentran en la Tabla 1. (Ver la hoja “No funcionales” en el libro anexo 2 Especificación de Requerimientos).

Los requerimientos no funcionales son aquellos que especifican cómo se proveen las funcionalidades del sistema.

## Priorización

Acorde con el mecanismo de priorización propuesto (sección 9.2.), se diligenció una tabla que lleva la priorización de todos los requerimientos. (Anexo 3 Priorización).

De esta manera, se puede determinar la importancia de los requerimientos en cuanto a el enfoque del trabajo de grado y recursos a disposición.

# Proceso de ingeniería de requerimientos

El proceso de ingeniería de requerimientos a seguir para la construcción de requerimientos del prototipo de Ane-Stent está descrito en el modelo de proceso de negocio del mismo nombre. (Anexo 4 Proceso de ingeniería de requerimientos)

En esta sección, se encuentra cada una de las fases descrita a mayor nivel de detalle.

## Mecanismos de priorización

Para la priorización de requerimientos, se hará uso de la plantilla que se puede apreciar en el anexo 3 Priorización. En la misma se puede detallar qué se debe indicar en cada celda y el resultado es una priorización semicuantitativa teniendo en cuenta los beneficios, penalizaciones, costos y riesgos. También se agrega un campo para una priorización cualitativa, en la que simplemente se asigna un valor alto, medio o bajo representando 1, 6.67 o 3.33 respectivamente, dependiendo de lo que crea quien esté usando la plantilla. Finalmente se promedian las prioridades y este es el resultado final.

Para obtener los valores calculados, se tiene en cuenta lo siguiente. (Wiegers, 2003)

* Valor total: Es la suma entre el beneficio y las penalizaciones relativas.
* Valor porcentual: Es el cociente entre el valor total por requerimiento y la suma de todos los valores totales.
* Costo porcentual: Es el cociente entre el costo relativo por requerimiento y la suma de todos los costos relativos.
* Riesgo Porcentual: Es el cociente entre el riesgo relativo por requerimiento y la suma de todos los riesgos relativos.
* Prioridad semicuantitativa: Es el cociente entre el valor porcentual y la suma del costo porcentual con el riesgo porcentual.

## Procesos de construcción de requerimientos

En la tabla 4, se explica de manera general las fases para llevar a cabo la construcción del requerimiento

|  |  |  |
| --- | --- | --- |
| **Fase** | **Objetivo** | **Herramienta** |
| **Obtención** | Listar los requerimientos | Documentos |
| **Verificar** | Verificar que los requerimientos cumplen con prácticas correctas de definición | Plantilla de verificación. |
| **Análisis** | Examinar lista de requerimientos | Plantilla de clasificación de requerimientos |
| Clasificar de requerimientos | Plantilla de clasificación de requerimientos |
| **Especificación** | Realización plantilla con descripción detallada de los requerimientos | Plantilla de especificación de requerimiento |

Tabla 4. Fases de construcción de requerimientos.

A continuación, se detallan los mecanismos y herramientas empleadas para el desarrollo de las fases.

### Obtención

Para esta fase, se tuvo en cuenta la investigación que hizo el equipo de trabajo en la plataforma de Thrombus VPH (Courbebaisse, 2014), lluvia de ideas del equipo de trabajo y las reuniones con el director de trabajo de grado Leonardo flores (sección 8.3. de plan de proyecto). Gracias a estas observaciones se logró obtener los requerimientos. (Anexo 5 Lista de requerimientos)

### Verificación

El proceso de validación es el siguiente paso del proceso. Este debe asegurar que los requerimientos cumplen con los estándares establecidos por el equipo, este proceso está descrito en la [sección 10 proceso de verificación](#_Proceso_verificación).  (Anexo 6 Verificación requerimientos)

### Análisis

Esta sección examina que se obtuvo previamente y se debe llenar la plantilla para clasificarlo. Para este se diseñó un formato para la clasificación de requerimiento (Anexo 7 Análisis de requerimientos). Se divide por tipos de requerimientos de hardware (H), software funcionales, estos especifican lo que debe hacer el sistema (F) y los software no funcionales (NF). Las especificaciones de la plantilla están descritas en la tabla 5.

|  |  |
| --- | --- |
| Campo | Descripción |
| ID requerimiento | Este es la identificación con el que se reconoce el requerimiento |
| Tipo | Puede ser de Software funcional, no funcional o hardware |

Tabla 5. Análisis requerimientos.

Al finalizar esta etapa se debe tener una idea general de los requerimientos que necesitan el proyecto de grado.

### Especificación

Esta etapa se realiza primero la priorización de los requerimientos (Anexo 3 Priorización requerimientos). Después, se diligencia la plantilla de especificación de requerimientos (Anexo 2 Especificación requerimientos) que contiene los siguientes campos en la tabla 6:

|  |  |
| --- | --- |
| **Campo** | **Descripción** |
| **ID** | Permite identificar un requerimiento de forma única |
| **Descripción** | Explica el requerimiento, en formato el “El sistema debe...” |
| **Prioridad** | Valor numérico resultante del cálculo en la priorización requerimientos. (Anexo 3 Priorización) |
| **Verificables** | Especifica si el requerimiento va a ser medido con una prueba y podrá determinarse si se cumple o no |
| **Dependencia** | Requerimientos que deben ser implementados antes de poder realizar el requerimiento en cuestión |
| **Anexo** | Documentación extra importante para la comprensión del requerimiento. |

Tabla 6. Campos de especificación de requerimientos.

# Proceso verificación

Todos los requerimientos propuestos, deberán ser sometidos a un proceso de validación en el cual, los miembros del equipo se encargarán de revisarlos para garantizar que cumplen con las “características de un buen requerimiento”. De esta forma, los requerimientos son sometidos a un proceso que evalúa criterios de aceptación que permite especificar requerimientos robustos.

A continuación, en la tabla 7, se explican a mayor detalle las características con las que deberá cumplir todo requerimiento especificado.

|  |  |
| --- | --- |
| Característica | Descripción |
| Completitud | El requerimiento no necesita aclaración extra, describe la capacidad y característica para satisfacer una necesidad. (ISO, 2011) |
| Correctitud | El requerimiento corresponde con una necesidad o restricción establecida en reglas de negocio. (Torres, 2016) (Ortíz, 2007) |
| Factibilidad | Es posible alcanzar el requerimiento, se encuentra dentro del alcance y no necesita de otros avances tecnológicos. (Torres, 2016) (ISO, 2011) |
| Necesidad | Su razón de ser está justificada por las reglas de negocio o fuente de recolección donde se obtuvo. (Torres, 2016) |
| Priorizabilidad | Tiene una importancia que lo distingue de otros requerimientos. (Torres, 2016) (Ortíz, 2007) |
| No ambigüedad | Tiene un único significado sin dar lugar otras interpretaciones. (Torres, 2016) (Ortíz, 2007) |
| Verificabilidad | Su cumplimiento es evaluable mediante algún proceso o métrica establecida. (Torres, 2016) (ISO, 2011) (Ortíz, 2007) |
| Independencia del diseño | Se puede alcanzar utilizando diferentes implementaciones. (Torres, 2016) (Ortíz, 2007) |
| No repetido | Todo requerimiento debe poseer información diferente respecto a necesidades y restricciones. (Torres, 2016) |
| Consistencia | No se presenta conflictos o contradicciones con otros requerimientos. (Torres, 2016) (Ortíz, 2007) |
| Conciso | El requerimiento tiene que estar centrado en las reglas de negocio y expresa la información necesaria de manera concreta y puntual. (Torres, 2016) (Ortíz, 2007) |
| Escrito en forma “Debe” | Este criterio permite que un requerimiento pueda ser escrito de tal forma que muestre el objetivo que intenta cumplir. (Torres, 2016) (ISO, 2011) |

Tabla 7. Características de un buen requerimiento.

La responsabilidad de evaluar los criterios en cada requerimiento recae sobre aquel integrante del equipo que no haya participado en el planteamiento inicial de este con el fin de que se pueda tomar una decisión crítica que permita dar retroalimentación completa a cada requerimiento planteado antes de su especificación formal.

Éste proceso se lleva a cabo mediante el diligenciamiento de una forma en la que el responsable debe decidir si en cada requerimiento se cumplen todos los criterios de aceptación. (Anexo 6 Verificación de requerimientos)

El proceso de verificación de requerimientos está descrito por el modelo de proceso de negocio anexo. (Anexo 8 Proceso verificación de requerimientos).

# Referencias

Courbebaisse, G. (25 de Junio de 2014). *A new model for the construction of virtual fully resolved flow-diverters | Thrombus-VPH*. Obtenido de Thrombus-VPH: http://thrombus-vph.eu/a-new-model-for-the-construction-of-virtual-fully-resolved-flow-diverters

Google LLC. (10 de Noviembre de 2017). *google/styleguide: Style guides for Google-originated open-source projects*. Obtenido de GitHub: https://github.com/google/styleguide

ISO. (2011). *Systems and software engineering - Life cycle processes-Requirements. Engineering.* ISO.

Ortíz, A. M. (2007). *SRS y Calidad de Requerimientos.* Obtenido de https://sophia.javeriana.edu.co/~lcdiaz/ingSw2007-1/calidadRequerimientos\_aOrtiz.pdf

TheServerSide. (8 de Febrero de 2013). *Contrasting distributed computing with concurrent programming*. Obtenido de TheServerSide.com: your Java Community discussing server side development: http://www.theserverside.com/feature/Contrasting-distributed-computing-with-concurrent-programming

Torres, M. (2016). Ingeniería de requerimientos. Bogotá: Pontifica Universidad Javeriana.

Wiegers, K. (2003). Beyond Funcionality. En *Software Requirements* (págs. 261-293). Microsoft.