



Software Plus



Monica Alvarez  
Juan Camilo Chafloque  
Sebastián Gutiérrez  
Laura Jimenez  
Santiago Palacios  
Sergio Posada

## **Smart Closet**

Software Requirements Specification

Versión 2.0

Ingeniería de Software

Pontificia Universidad Javeriana  
Facultad de Ingeniería  
Departamento de Ingeniería de Sistemas  
2020-1

## 1. Historial de cambios

<b>Versión</b>	<b>Fecha de Cambio</b>	<b>Descripción</b>	<b>Sección o Artefacto Afectado</b>	<b>Responsable</b>
0.1	26/03/2020	Creación del documento y Creación de Portada  Realización del modelo de dominio y su documentación	Portada, Historial de cambios, 7.1	Sebastián Gutiérrez, Laura Jimenez, Monica Alvarez
0.2	27/03/2020	Descripción de la perspectiva del producto	Sección 7.2	Monica Alvarez
0.3	27/03/2020	Descripción de las interfaces del sistema.	Sección 7.2.1	Sebastián Gutiérrez
0.4	27/03/2020	Descripción de operaciones del software.	Sección 7.2.2	Laura Jimenez
0.5	30/03/2020	Realización del diagrama de casos de uso.	Sección 7.3	Monica Alvarez
0.6	30/03/2020	Documentación de los casos de uso.	Sección 7.3	Monica Alvarez, Sebastian Gutierrez, Laura Jimenez
0.7	31/03/2020	Documentación de los casos de uso.	Sección 7.3	Monica Alvarez, Sebastian Gutierrez, Laura Jimenez, Santiago Palacios
0.8	01/04/2020	Descripción de los roles y usuarios, descripción de los requisitos no funcionales.	Sección 7.4, 7.5	Monica Alvarez, Sebastian Gutierrez, Laura Jimenez
0.9	05/04/2020	Documentación requisitos funcionales.	Sección 8.1	Sebastian Gutierrez, Laura Jimenez, Santiago Palacios
1.0	06/04/2020	Características del producto de software.	Sección 8.1	Sebastian Gutierrez, Laura Jimenez
1.1	06/04/2020	Descripción de los requisitos de desempeño.	Sección 8.2	Sebastian Gutierrez

1.2	06/04/2020	Descripción de los atributos de calidad.	Sección 8.3	Monica Alvarez, Laura Jimenez
1.3	07/04/2020	Descripción de la ingeniería de procesos.	Sección 9.1	Sebastian Gutierrez, Monica Alvarez
1.4	08/04/2020	Descripción del levantamiento de requisitos.	Sección 9.2	Monica Alvarez, Santiago Palacios, Sebastian Gutierrez, Laura Jimenez
1.5	08/04/2020	Diagrama y descripción de validación de requisitos.	Sección 10	Laura Jimenez, Monica Alvarez, Sebastian Gutierrez
1.6	09/04/2020	Administración de requisitos	Sección 9.3	Laura Jimenez, Monica Alvarez, Sebastian Gutierrez
1.7	09/04/2020	Documentación requisitos no funcionales.	Sección 8.1	Laura Jimenez
1.8	09/04/2020	Escribir resumen del documento.	Sección 2.0	Sergio Posada
1.9	11/04/2020	Introducción	Sección 6	Santiago Palacios
2.0	14/04/2020	Revisión y corrección del documento	Todas las secciones	Monica Alvarez, Juan Camilo Chafloque, Sebastian Gutierrez, Laura Jimenez, Santiago Palacios, Sergio Posada

*Tabla 1. Historial de cambios*

## 2. Resumen

Este documento tiene como objetivo principal describir todas las funcionalidades de la aplicación web Smart-Closet según las directrices dadas por la profesora Anabel Montero. Además de esto, el documento provee los requisitos sobre los cuales el equipo de trabajo Software Plus trabajó, junto a sus respectivas especificaciones, con el fin de lograr una buena calidad y optimización a la hora de llevar a cabo el desarrollo de esta segunda parte del proyecto semestral.

Respecto al contenido de este documento se detallará la funcionalidad del sistema en su totalidad desde la perspectiva de los usuarios. Por otra parte, se describirán en profundidad todos los componentes que hacen parte del sistema, sus operaciones, características de los usuarios, entre otros.

### 3. Tabla de contenidos

6.	Introducción .....	1
7.	Descripción global .....	1
7.1.	Modelo de dominio .....	1
7.1.1.	Diagrama de clases.....	2
7.1.2.	Documentación diagrama del modelo del dominio .....	2
7.2.	Perspectiva del producto.....	5
7.2.1.	Interfaces del sistema.....	6
7.2.2.	Operaciones .....	7
7.2.2.1.	Modos de operación.....	7
7.2.2.1.1.	Modo usuario.....	7
7.2.2.2.	Periodos de actividad e inactividad .....	7
7.2.2.3.	Procesos de recuperación .....	7
7.2.2.4.	Función de soporte a procesamiento de datos.....	8
7.3.	Funciones del producto .....	8
7.4.	Características del usuario.....	9
7.5.	Requisitos no funcionales .....	10
7.6.	Suposiciones y dependencias .....	11
7.6.1.	Suposiciones .....	11
7.6.2.	Dependencias .....	12
8.	Requisitos específicos .....	12
8.1.	Características del producto de software .....	12
8.2.	Requisitos de desempeño.....	13
8.2.1.	Requisitos estáticos .....	13
8.2.2.	Requisitos dinámicos .....	14
8.3.	Atributos del sistema de software .....	14
8.4.	Requisitos de la base de datos .....	15

9.	Proceso de ingeniería de requisitos .....	15
9.1.	Planeación .....	16
9.2.	Levantamiento.....	17
9.3.	Administración de requisitos.....	18
10.	Proceso de verificación y validación.....	18
11.	Referencias .....	22

#### 4. Lista de figuras

Ilustración 1.	Diagrama de clases.....	2
Ilustración 2.	Diagrama de casos de uso.....	9
Ilustración 3.	Proceso de ingeniería de requisitos. ....	17
Ilustración 4.	Proceso de levantamiento de requisitos .....	18
Ilustración 5.	Diagrama de flujo para validar un requisito .....	20
Ilustración 6.	Diagrama de flujo para verificar un requisito. ....	21

#### 5. Lista de tablas

Tabla 1.	Historial de cambios .....	II
Tabla 2.	Elementos del dominio "Usuario" .....	3
Tabla 3.	Elementos del dominio "Armario" .....	3
Tabla 4.	Elementos del dominio "Prenda" .....	4
Tabla 5.	Elementos del dominio "Accesorio" .....	4
Tabla 6.	Elementos del dominio "Zapato".....	4
Tabla 7.	Elementos del dominio "Superior".....	4
Tabla 8.	Elementos del dominio "Inferior".....	5
Tabla 9.	Elementos del dominio "Color" .....	5
Tabla 10.	Elementos del dominio "Atuendo" .....	5
Tabla 11.	Interfaces del sistema .....	7
Tabla 12.	Descripción de las características del usuario .....	10
Tabla 13.	Clasificación de requisitos no funcionales .....	11
Tabla 14.	Requisito funcional agregar prenda.....	13
Tabla 15.	Atributo de calidad A3.....	15

## **6. Introducción**

El desarrollo del presente documento tiene como finalidad presentar la documentación relacionada a la especificación de requisitos del software (SRS) de la empresa Software Plus, para el desarrollo de su producto Smart Closet. Este último es un software que busca facilitar a los usuarios una forma innovadora para decidir qué tipo de prendas usarán en un día determinado según sus preferencias. El software tiene un conjunto de reglas de moda que le permite recomendar atuendos adecuados, por otra parte, para generar un atuendo también puede ser tenidos en cuenta los estándares establecidos por el usuario, el sistema tendrá en cuenta también qué tipo de condiciones climáticas se van a dar en un día, y con base en esto, Smart Closet brindará la mejor opción para el usuario.

El principal objetivo del SRS es la de proveer una descripción completa del comportamiento del sistema, por lo tanto, este documento comienza con una descripción global donde se hace un primer acercamiento a los requisitos y el levantamiento de los mismos, se especifica el modelo de dominio, el contexto y el origen del producto junto a las interfaces, funcionalidades y operaciones principales del producto, posteriormente en la sección de requisitos específicos se describen de manera más precisa algunas subsecciones de la sección anterior haciendo especial énfasis en las características del producto y los requisitos de software, de desempeño y de la base de datos.

Además de lo descrito anteriormente, el SRS también da una explicación completa del proceso de levantamiento de requerimientos, su planeación, su administración y la posterior verificación y validación de los mismos.

## **7. Descripción global**

En esta sección se identificarán y se hará una primera aproximación a los requisitos y factores generales que afectan el producto. Estos se irán especificando a lo largo del documento de manera más detallada ya que son aquellos elementos que modelan las necesidades del cliente frente al proyecto que se está realizando.

### **7.1. Modelo de dominio**

En esta sección se encontrarán las representaciones de las clases conceptuales del modelo de negocio de Smart Closet mediante el Diagrama del modelo del dominio junto con la documentación relevante de cada entidad, con el fin de dar una aproximación del sistema a generar.

### 7.1.1. Diagrama de clases

A continuación, se puede ver el diagrama de clase en la *Ilustración 1. Diagrama de clases* el cual representa las relaciones de las entidades del modelo de negocio de Smart closet.

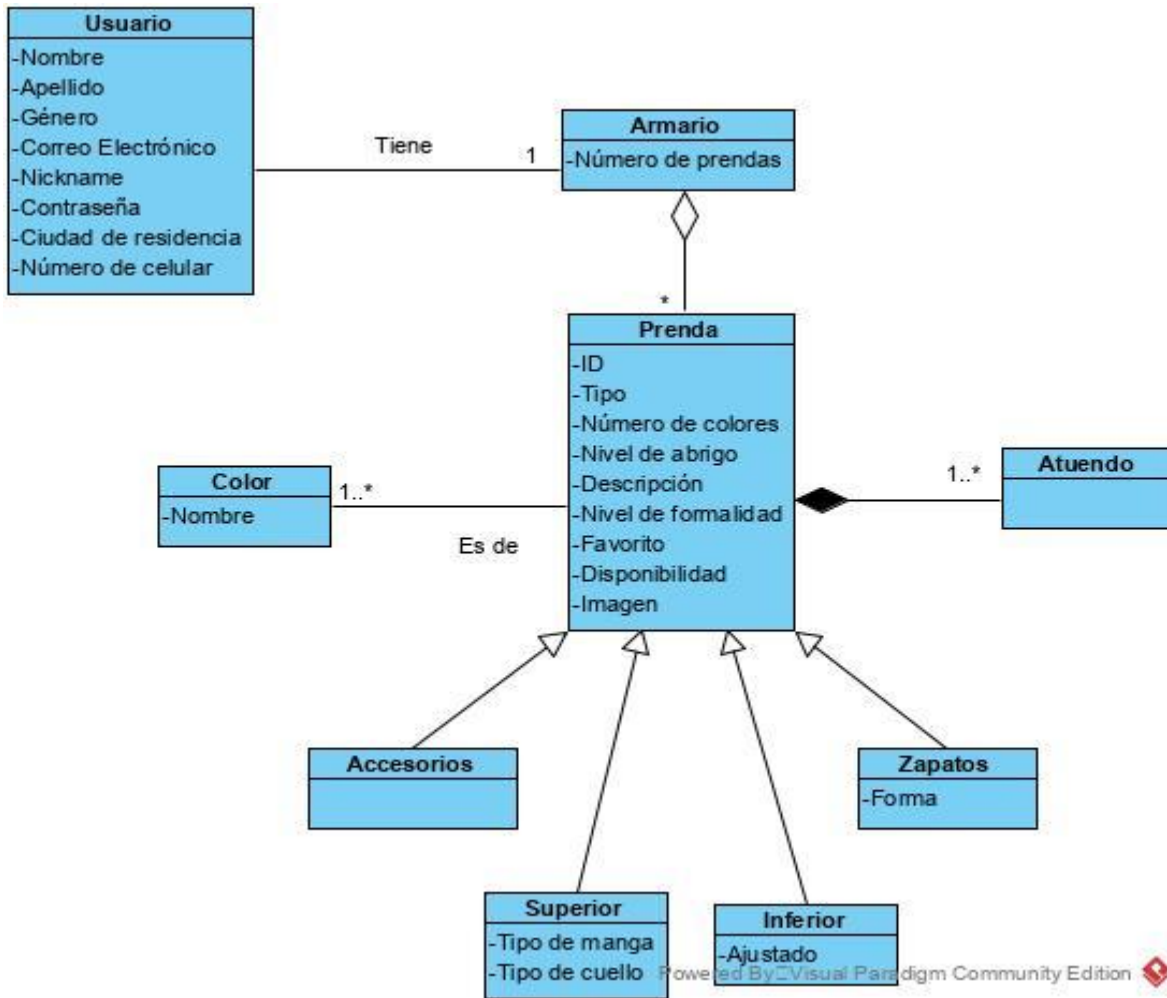


Ilustración 1. Diagrama de clases

### 7.1.2. Documentación diagrama del modelo del dominio

Se pueden visualizar las tablas de las entidades que fueron abstraídas del dominio del sistema y la descripción detalla de cada una.

ID	00	Elemento del dominio	Usuario
----	----	----------------------	---------

<b>Descripción</b>	Representa un cliente que utiliza la aplicación web para gestionar su closet virtual y los atuendos que se generan a partir de sus necesidades y prendas disponibles
<b>Atributos</b>	
<b>Nombre atributo</b>	
Nombre	
Apellido	
Género	
Correo electrónico	
Número de celular	
Nickname	
Contraseña	
Ciudad de residencia	

*Tabla 2. Elementos del dominio "Usuario"*

ID	01	Elemento del dominio	Armario
Descripción	Entidad que representa el conjunto total de prendas que posee el usuario.		
Atributos			
Nombre atributo			
Número de prendas			

*Tabla 3. Elementos del dominio "Armario"*

ID	02	Elemento del dominio	Prenda
Descripción	Elemento que representa un artículo con el que se viste el usuario		
Atributos			
Nombre atributo			
ID			
Tipo			
Cantidad de colores			
Nivel de abrigo			
Descripción			



Nivel de formalidad
Favorito
Disponibilidad
Imagen

*Tabla 4. Elementos del dominio "Prenda"*

ID	03	Elemento del dominio	Accesorio
Descripción	Tipo de prenda que se encarga de complementar el atuendo de un usuario.		
Atributos			
Nombre atributo			

*Tabla 5. Elementos del dominio "Accesorio"*

ID	04	Elemento del dominio	Zapato
Descripción	Tipo de prenda que protege los pies de un usuario.		
Atributos			
Nombre atributo			
Forma del zapato			

*Tabla 6. Elementos del dominio "Zapato"*

ID	05	Elemento del dominio	Superior
Descripción	Representa las prendas que se usan en la parte superior del cuerpo (torso)		
Atributos			
Nombre atributo			
Tipo de manga			
Tipo de cuello			

*Tabla 7. Elementos del dominio "Superior"*

ID	06	Elemento del dominio	Inferior
----	----	----------------------	----------

<b>Descripción</b>	Representa las prendas que se usan en la parte inferior del cuerpo
<b>Atributos</b>	
<b>Nombre atributo</b>	
Ajustado	

*Tabla 8. Elementos del dominio "Inferior"*

ID	07	Elemento del dominio	Color
Descripción	Cualidad visual que puede tener una prenda, útil para distinguir características específicas de cada prenda		
Atributos			
Nombre atributo			
Nombre			

*Tabla 9. Elementos del dominio "Color"*

ID	08	Elemento del dominio	Atuendo
Descripción	Combinación de prendas relacionadas según los criterios establecidos		
Atributos			
Nombre atributo			
Prendas			

*Tabla 10. Elementos del dominio "Atuendo"*

## 7.2. Perspectiva del producto

Smart Closet nace gracias a la necesidad que tienen las personas de vestir de manera adecuada sus prendas que poseen, ya que a éstas se les dificulta en ocasiones seleccionar el mejor atuendo para días especiales, compromisos formales o simplemente casuales. Dichas personas pasan desapercibidos factores cruciales que pueden hacer que sus atuendos luzcan con mayor afinidad con respecto a sus diseños y, asimismo, sean las más adecuadas al día. Por tanto, Smart Closet se proyecta como una aplicación web que ayuda al usuario a seleccionar el mejor atuendo a partir de condiciones de moda, criterios usuario y condiciones del día.

A pesar de que en la actualidad existen diversos productos que buscan solucionar el mismo problema, Smart Closet se concibe como un producto nuevo con un valor agregado al usuario que no solo selecciona un atuendo a partir de un conjunto de

prendas disponibles, sino que también recomienda el mejor atuendo basado en condiciones climáticas y de tiempo, tomando criterios pertinentes a la hora de evaluar qué prendas se sitúan mejor a las condiciones sin perder la afinidad entre las prendas, permitiendo que no solamente el atuendo combine de la mejor manera, sino que también beneficie al usuario en su día a día.

Además del fin general del producto y del valor agregado, Smart Closet ofrece otras funciones que permiten al usuario interactuar con mayor comodidad con el programa, pudiendo sacar con mayor provecho lo que la aplicación tiene para ofrecer. Entre sus funciones se pueden encontrar:

- Gestión total de las prendas que están en el armario virtual.
- Inhabilitar prendas para que no sean tenidas en cuenta en la selección del atuendo.
- Selección de prendas y atuendos favoritos.
- Compartir los atuendos.

### 7.2.1. Interfaces del sistema

En esta sección se define y se especifican las interfaces que interactúan en el sistema Smart Closet. Dado que se trata de un producto nuevo las interfaces que se describen son propiamente elaboradas por Software Plus y no se basan de interfaces ya elaboradas anteriormente por terceros. A continuación, en la *Tabla 11. Interfaces con el sistema* se puede visualizar las interfaces con el sistema y su descripción.

Interfaz	Descripción
Usuario	La aplicación web cuenta con un interfaz de usuario por el cual este podrá hacer uso de todas las funciones del programa desde el ingreso de datos, la navegación entre ventanas que cada función del programa provee al usuario.
Hardware	Se debe contar con un hardware que permita la conexión a internet ya sea por medio de conexiones LAN, Wi-Fi, o servicios de red 3G o 4G para la comunicación entre los dispositivos que hacen uso del programa.
Base de datos (Spring H2DataBase)	Se tiene en cuenta una interfaz de persistencia, el cual será brindado por Spring. Para el caso del paso de información entre la aplicación y <i>H2DataBase</i> se concibe que los datos nunca serán descargados en el programa por esto se mantiene un contacto permanente con Spring a través de una constante sincronización de datos.
Interfaces externas	Además de las interfaces mencionadas, el sistema cuenta con unas interfaces externas que se comunican con las funciones del programa que permite el funcionamiento óptimo del programa. Dichas interfaces corresponden a otros sistemas como un motor de búsquedas de imágenes para la obtención de las ilustraciones de las prendas, interfaces del API que

	provee el clima y el interfaz con el sistema de Whatsapp para poder compartir los atuendos.
--	---

*Tabla 11. Interfaces del sistema*

## **7.2.2. Operaciones**

### **7.2.2.1. Modos de operación**

#### **7.2.2.1.1. Modo usuario**

Este es el único modo con el que cuenta la aplicación Smart Closet al cual se puede ser accedido mediante cualquier dispositivo electrónico con un navegador y acceso a internet. Este modo se encuentra disponible para aquellos usuarios que ya se encuentran registrados en la aplicación, los cuales podrán acceder a la totalidad de las funcionalidades ofrecidas por la misma.

#### **7.2.2.2. Periodos de actividad e inactividad**

Smart Closet será una aplicación que se mantendrá activa 24 horas al día durante 7 días a la semana, sin embargo, mensualmente presentará una interrupción de 2 horas con el fin de realizar revisión y mantenimiento.

Por otra parte, Smart Closet no cuenta con ninguna manera de actualización directa o con algún modo tipo administrador que permita realizar modificaciones, es por esto que cada vez que esta necesite ser actualizada o se requiera lanzar una nueva funcionalidad será necesario la inactividad durante el período de tiempo que se tarde llevar a cabo dichas tareas.

#### **7.2.2.3. Procesos de recuperación**

El proceso de recuperación se realizará con una herramienta de línea de comandos de PostgreSQL llamada pg\_dump la cual permite hacer un respaldo de bases de datos [1]. En principio se buscará que éste se haga de manera automática y con una frecuencia semanal. Sin embargo, habrá un responsable de revisar que esta tarea se esté cumpliendo de manera correcta y en caso de que ocurra un error, esta persona encargada hará el respaldo de manera manual.

Una vez realizada la copia de seguridad, será almacenada en una carpeta de archivos de Google Cloud Storage con su respectiva fecha de creación. Además, se mantendrán almacenados los respaldos de a lo sumo un mes con el fin de ahorrar espacio en el servicio de almacenamiento.

Esta actividad se realizará los lunes en la madrugada (2:00 am) puesto que, se espera que en ese momento la cantidad de usuarios que estén haciendo uso del

sistema sea mínima o nula con el fin de evitar errores a la hora de realizar el respaldo.

#### **7.2.2.4. Función de soporte a procesamiento de datos**

El procesamiento de datos general se realizará mediante la lógica de negocio del sistema y la interacción con la base de datos. Sin embargo, se requiere el uso de un sistema externo para obtener el clima actual, de esta manera se hará uso de OpenWeatherMap y GoogleMaps. Para el motor de búsqueda de las imágenes asociadas a las prendas se hará uso de GoogleSearchEngine y finalmente está por decidirse que sistema externo se usará para la opción de compartir el atuendo a través de WhatsApp.

Por una parte las actualizaciones de versiones del sistema para mejorar desempeño, la agregación de nuevas funcionalidades, el mantenimiento general de la aplicación o en caso tal que el usuario experimente algún problema con la misma, serán realizadas por el equipo programador como ya fue especificado, por otro lado, todas estas tareas de soporte de los sistemas externos no son responsabilidad de Software Plus, sino, al ser un servicio solicitado, el proveedor es quien las realiza.

### **7.3. Funciones del producto**

En esta sección se podrá visualizar una descripción de las funciones del producto que ofrece Software Plus. Se detallará las funciones que proveerá Smart Closet y la especificación de estas, con el fin de que se pueda comprender en su totalidad lo que el sistema ofrece y que se tenga un acercamiento de lo que el usuario dispondrá finalmente.

A continuación, se puede ver el diagrama de Casos de Uso del sistema Smart Closet en la *Figura 2. Diagrama de Casos de Uso* en donde se detalla la interacción que tiene el usuario con las diferentes funcionalidades del sistema y como estas se comportan entre sí. Adicionalmente se puede ver en el [Anexo 1. Documentación Casos de uso](#) la especificación de cada funcionalidad del sistema, describiendo los actores, las condiciones, los escenarios y el flujo de estos mismos.

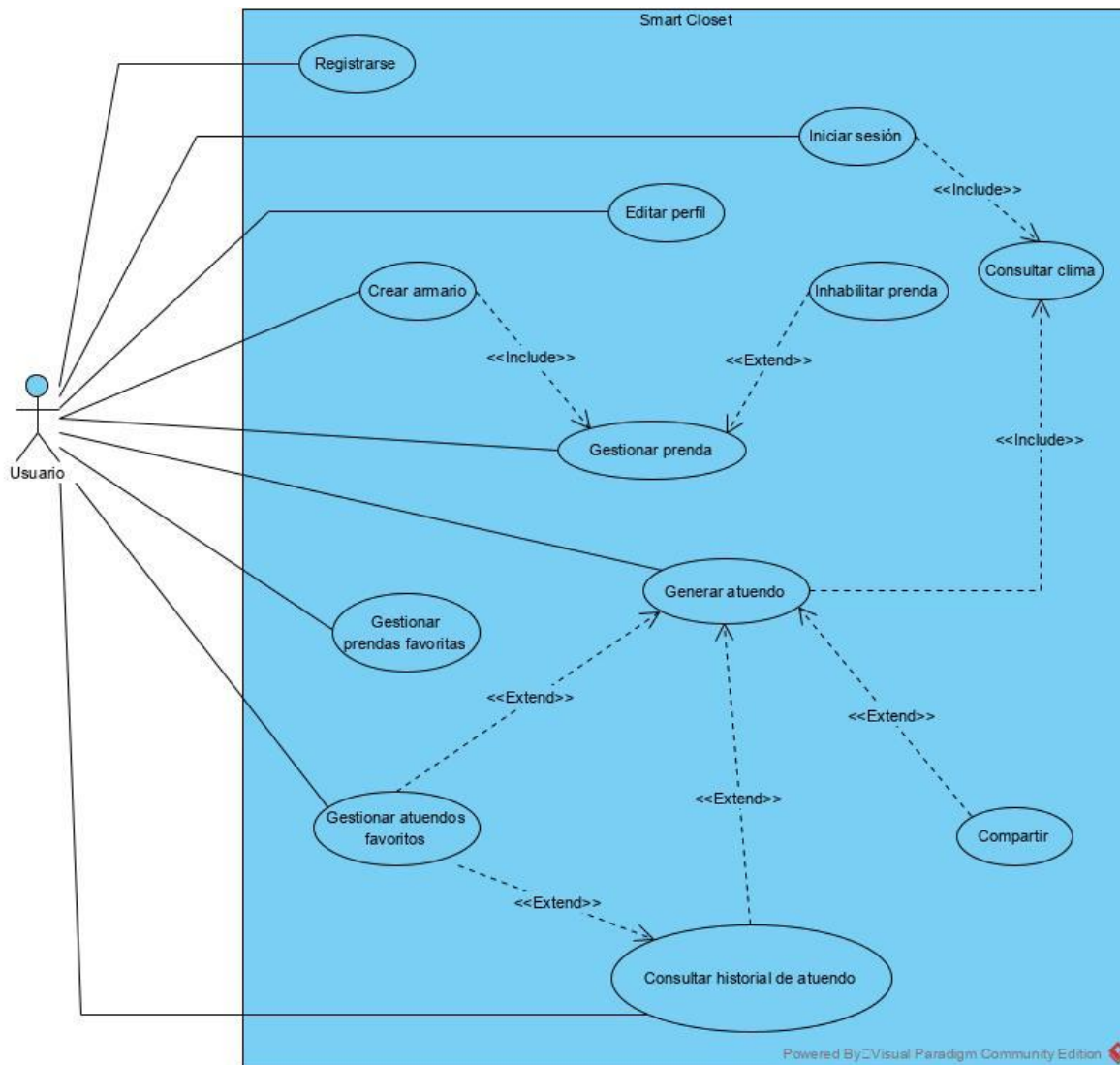


Ilustración 2. Diagrama de casos de uso

#### 7.4. Características del usuario

Dado que la aplicación Smart Closet está dirigida a todo tipo de público a partir de los 10 años de edad, se tiene sólo un tipo de usuario. A continuación, se detallan los privilegios y las funcionalidades a las que éste podrá acceder.

Características del usuario	Descripción
Nivel de seguridad o de privilegios	<p>El usuario puede:</p> <ul style="list-style-type: none"> <li>→ Crear una cuenta con sus datos personales.</li> <li>→ Acceder a su perfil cada vez que lo necesite.</li> </ul>

	<p>→ Editar los datos personales proporcionados a la hora de hacer el registro.</p> <p>→ Contener la información de su ropa de manera privilegiada en un armario virtual.</p> <p>→ Ingresar sus prendas de vestir haciendo uso de un motor de búsqueda o cargando a la aplicación una fotografía de dicha prenda.</p> <p>→ Caracterizar cada una de las prendas que ingrese.</p> <p>→ Visualizar el clima de la ciudad con la que hizo el registro.</p> <p>→ Obtener recomendaciones de atuendos según sus necesidades.</p> <p>→ Marcar, visualizar o eliminar las prendas que más le gusten como favoritas</p> <p>→ Marcar, visualizar o eliminar los atuendos que más le gusten como favoritos.</p> <p>→ Ver los últimos 4, 8 o 12 atuendos generados por la aplicación.</p> <p>→ Marcar las prendas como inhabilitadas según su necesidad.</p> <p>→ Compartir el atuendo generado por la aplicación a través de WhatsApp.</p>
Roles	El único rol del usuario es ser administrador de su propio armario
Nivel de estudios o experiencia técnica	Primaria y conocimientos básico de manipulación de páginas web.
Frecuencia de uso	El usuario podrá frecuentar la aplicación web diario. Lo ideal es que el usuario disponga todos los días de la aplicación para que pueda vestirse cada día según la recomendación que el sistema brinda.

*Tabla 12. Descripción de las características del usuario*

## 7.5. Requisitos no funcionales

En esta sección se describen los requisitos no funcionales del aplicativo web Smart Closet, es decir, se especifican aquellas restricciones que no dependen intrínsecamente en las reglas del negocio, sino que son consecuentes de factores externos como aspectos legales, condiciones de hardware, componentes de comunicación, sistemas alternos y reglas generales como la tolerancia a fallos y el tiempo de respuesta de las funcionalidades. A continuación, se puede visualizar los requisitos no funcionales identificados con su respectiva clasificación según el modelo FURPS+.

<b>Requisito</b>	<b>FURPS+</b>
La aplicación debe funcionar en cualquier navegador web.	<i>Implementación (+)</i>
El sistema debe soportar el protocolo HTTP.	<i>Facilidad de uso</i>
El sistema debe ser lo suficientemente intuitivo para que el usuario no necesite de una capacitación.	<i>Facilidad de uso</i>
El sistema debe tener un tiempo de respuesta menor a 5 segundos en todas sus funcionalidades.	<i>Rendimiento</i>
El sistema debe notificarle al usuario cualquier solicitud que haya cancelado.	<i>Fiabilidad</i>
El sistema debe consumir menos de 50 MB de memoria RAM.	<i>Implementación (+)</i>
El sistema no debe ocupar espacio de almacenamiento en el dispositivo del usuario.	<i>Implementación (+)</i>
Las interfaces del sistema deben funcionar para cualquier resolución de pantalla.	<i>Implementación (+)</i>
El sistema debe ser desarrollado bajo herramientas, API's, imágenes, librerías y licencias gratuitas.	<i>Legales (+)</i>
El sistema debe ser codificado en Java, Typescript, HTML y CSS.	<i>Implementación (+)</i>
El servidor debe tener al menos una disponibilidad del 98% al año.	<i>Fiabilidad</i>
El sistema debe funcionar con un máximo de 100 usuarios conectados simultáneamente.	<i>Rendimiento</i>

*Tabla 13. Clasificación de requisitos no funcionales*

## **7.6. Suposiciones y dependencias**

### **7.6.1. Suposiciones**

- Los dispositivos en los cuáles los usuarios disponen deben tener conexión a Internet.
- Los dispositivos deben tener cualquier navegador web que soporte el sistema.
- Las API's y la base de datos Spring que utiliza el sistema funcionarán correctamente con este y estarán disponibles todo el tiempo exceptuando los momentos de actualización.
- El dispositivo dispone de un GPS para la funcionalidad de consultar el clima.
- El usuario posee una gran variedad de ropa para que el sistema no genere atuendos triviales o repetitivos.



- Los usuarios de la aplicación web estarán dispuestos a proveer información personal y a otorgar permisos para acceder a funcionalidades específicas de sus dispositivos.

#### **7.6.2. Dependencias**

- La velocidad de la aplicación web depende parcialmente de la capacidad de procesamiento del dispositivo y la calidad de internet que su proveedor le otorgue.
- La base de datos Spring y el framework Angular no deben hacer cambios significativos que modifiquen de manera significativa el funcionamiento y/o desarrollo del proyecto.
- La variedad de atuendos generados depende de la variedad y disponibilidad de las prendas del usuario.
- El dato obtenido a través API “Open Weather Map” es consistente con el clima de la ciudad para generar un atuendo que cumpla con las especificaciones.

### **8. Requisitos específicos**

Una vez identificados los requisitos se procede a especificarlos de manera detallada, clara y sencilla tal que sea entendido por los dueños del negocio, pero sin dejar de lado el lenguaje técnico y profesional usado en el marco de la ingeniería.

Para esto los requisitos se documentarán haciendo uso de la tabla Volere, un formato que permite especificar los requisitos de manera que se cumpla con las doce características que debe cumplir un requisito para que sea válido.

#### **8.1. Características del producto de software**

Los requisitos funcionales de un sistema son aquellos que especifican las tareas, funciones o comportamientos que éste llevará a cabo a medida que se cumplan ciertas condiciones, con el fin de que se acepten entradas, se procese información y se generen salidas válidas en el sistema [2]. Dichos requisitos fueron definidos por Software Plus cuando la idea de Smart Closet estaba incursionando, no obstante, dichos requisitos han ido evolucionando conforme el proyecto se desarrolla.

En su totalidad se identificaron 18 requisitos funcionales, de los cuales se generaron los casos de uso que se mencionaron en la sección [7.3 Funciones del producto](#). La documentación de estos requisitos se hizo con la plantilla Volere, a manera de ejemplo en la *Tabla 14. Requisito funcional de agregar prenda* se puede detallar la forma en que los requisitos funcionales fueron documentados, especificando los

campos más importantes que son la descripción y la razón, cumpliendo con las características de un requisito bien descrito. Todos los requisitos funcionales se encuentran documentados y explicados en el [Anexo 2. Documentación de requisitos \(Sección 1\)](#).

<b># Requisito</b>	03	<b>Tipo de requisito</b>	Funcional	<b>Casos de uso asociados</b>	Gestionar prenda
<b>Descripción</b>	El sistema es capaz de <b>agregar</b> nuevas prendas. Los datos que se persisten son: Una imagen, el tipo, número de colores, nivel de abrigo, diseño, nivel de formalidad.				
<b>Razón</b>	Es necesario poseer prendas en el armario virtual para que de ese modo el sistema pueda disponer de éstas en las demás funcionalidades como la principal que es generar un atuendo.				
<b>Autor</b>	Laura Jimenez				
<b>Criterio de medición</b>	Se verifica que la cantidad de prendas que existen en el armario haya aumentado coherentemente con la cantidad de prendas ingresadas.				
<b>Prioridad</b>	Alto	<b>Módulo asociado</b>		Componente Prenda	
<b>Versión</b>	3.0	<b>Fecha</b>		29/02/2020	

Tabla 14. Requisito funcional agregar prenda

## 8.2. Requisitos de desempeño

En esta sección se describen los requisitos de desempeño que debe poseer el aplicativo web Smart Closet. Estos requisitos se dividirán en estáticos y dinámicos. Los servicios y capacidades de la versión gratuita de Spring Boot [3] permiten determinar con mayor precisión los requisitos de desempeño del sistema, ya que esta aplicación dispondrá de los servicios de Spring como servidor para ponerse en ejecución.

A continuación, se enlista los requisitos de desempeño, sin embargo, la documentación se encuentra en el [Anexo 2. Documentación de requisitos \(Sección 2\)](#) cada requisito es especificado con la plantilla Volere.

### 8.2.1. Requisitos estáticos

- El sistema debe correr en cualquier navegador web de cualquier sistema operativo.
- El sistema debe soportar 100 usuarios concurrentes.
- El sistema debe soportar transacciones de máximo 30 MB.
- El sistema consume a lo sumo 50 MB de memoria RAM.
- El sistema debe funcionar en cualquier tamaño de ventana.

### 8.2.2. Requisitos dinámicos

- El sistema debe soportar 100 transacciones concurrentes.
- El sistema debe tener un tiempo de respuesta menor de 5 segundos en sus solicitudes.
- El sistema aborta las solicitudes que hizo el usuario que sobrepasen el tiempo de 5 segundos.
- El sistema debe informar al usuario de aquellas solicitudes que fueron abortadas.
- El sistema tiene un tiempo de repuesta máximo de 60 segundos por transacción.
- El sistema debe tener una disponibilidad del 98% mensual de tiempo.

### 8.3. Atributos del sistema de software

Esta sección se centra en describir todos los aspectos de calidad de Smart Closet, para definir los atributos del sistema el equipo de Software Plus se apoyó en las facilidades que ofrece usar Spring Boot para el desarrollo del proyecto, también se tuvo en cuenta la arquitectura que propone Angular, dado que es la herramienta principal para la codificación de la aplicación.

La identificación de los atributos del sistema estuvo liderada por el Arquitecto de Software y la Responsable de Calidad, posteriormente se realizó la documentación pertinente haciendo uso de la plantilla de Volere con unas ligeras modificaciones, este proceso de documentación estuvo a cargo de la Jefe de Documentación en compañía con la Responsable de Calidad.

A continuación, se observa un ejemplo de la documentación resultante de este proceso. Los demás atributos del sistema se encuentran documentados en el [Anexo 2. Documentación de requisitos \(Sección 3\)](#).

# Atributo	A3	Seguridad	Requisitos asociados	
<b>Descripción</b>		El método de seguridad inicial es la creación de una cuenta para cada usuario que garantice un espacio privilegiado y exclusivo para la información de cada uno de ellos al cual podrá ser accedido a través de un nickname o el correo electrónico y una contraseña. Por otro lado, Spring ofrece servicio de cifrado de datos, de autenticación bajo protocolos como LDAP [2] y modificación de permisos las cuales son acciones que apoyan la seguridad del software desde la programación del servidor.		
<b>Razón</b>		Es necesario garantizar la seguridad de la información que el usuario principal ingresa en el software, especialmente teniendo		

	en cuenta que para registrarse usa algunos datos personales y datos sensibles como es el género.		
<b>Autor</b>	Laura Jimenez		
<b>Versión</b>	1.0	<b>Fecha</b>	06/04/2020

Tabla 15. Atributo de calidad A3

#### 8.4. Requisitos de la base de datos

Una vez especificado los requisitos funcionales, de desempeño y de calidad, se procede a describir los requerimientos que la base de datos que dispone Smart Closet. A continuación, se describe estos requerimientos teniendo en cuenta la documentación de las características de la base de datos que provee Spring: H2 Data Base [4]. Se especifica el rendimiento de la base de datos, el funcionamiento de las consultas, las restricciones, los tipos de datos, entre otras características relevantes.

- La base de datos debe contener tipos de datos como *Int, Boolean, TinyInt, SmallInt, BigInt, Identity, Decimal Double, Real, Time, Date, TimeStamp, Binary, Varchar, Char, Blob, Clob, Uuid, Array, Enum, Geometry, Jason* e *Interval*.
- La base de datos debe ser compatible con otras bases de datos como *HSQLDB, MySQL* y *PostgreSQL*.
- La base de datos debe tener tipos de consultas SQL y el acceso a los datos debe ser por medio de *JDBC* u *ODBC*.
- La base de datos debe soportar una indexación de datos de tipo *b-tree, tree* y *hash*.
- La base de datos debe soportar una indexación de múltiples dimensiones.
- La base de datos debe tener un tiempo de respuesta menor a 4 segundos en cualquier consulta.
- La base de datos debe soportar máximo 50 transacciones por segundo.
- La base de datos debe consumir menos 17 MB de memoria RAM.
- La base de datos debe cumplir con las propiedades *ACID* para las transacciones.
- La base de datos debe tener una frecuencia de acceso variado puesto que posee múltiples conexiones abiertas para acceso de datos *JDBC*.

#### 9. Proceso de ingeniería de requisitos

En esta sección se detallará a profundidad todo el proceso de ingeniería de requisitos que Software Plus planeó e implementó para las funcionalidades de Smart Closet. Se hará mención a los pasos y actividades que se llevaron a cabo como también la trazabilidad de cada uno de los requisitos identificados a través de diagramas y tablas.

## 9.1. Planeación

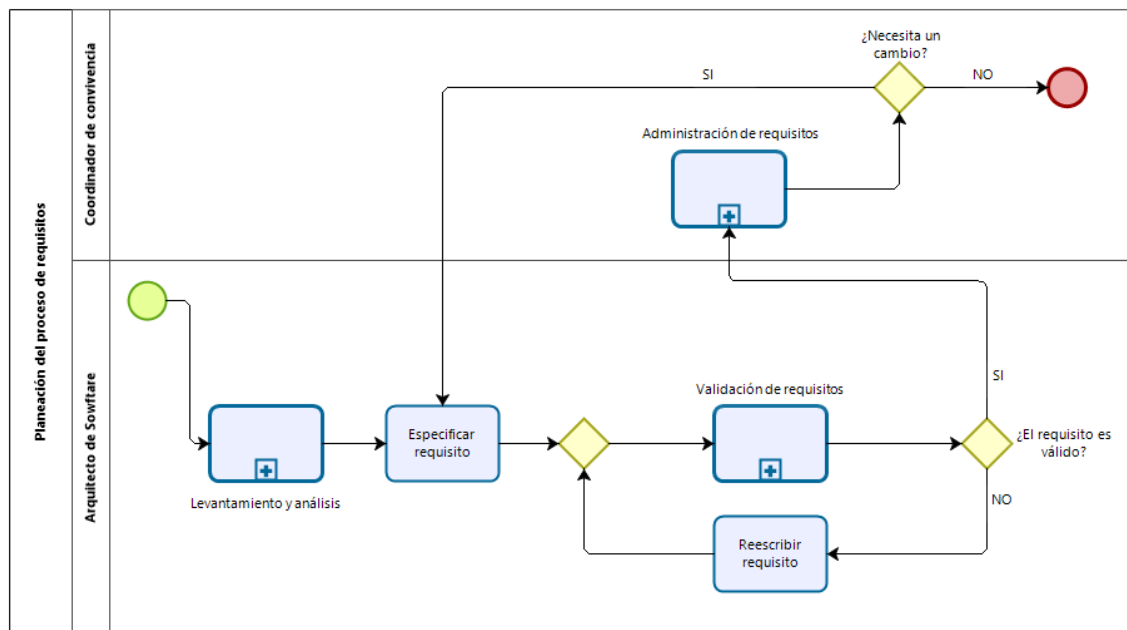
Tal como se especificó la planeación de la administración de los requerimientos en la *sección 10.1.1. Planeación de administración de requerimientos* del SPMP, este proceso consta de cuatro tareas importantes: identificar los requisitos más importantes, procesos de administración de cambios, políticas de seguimiento y herramientas de apoyo.

Los encargados para la planeación y puesta en marcha del proceso de requisitos serán el Coordinador de Convivencia y el Arquitecto de software. También se dispone del director del proyecto para la especificación de los requisitos de desempeño puesto que él es el que precisa con mayor certeza datos relevantes para ese tipo de requerimientos, tales como la tasa de disponibilidad y rendimiento del software. Estos encargados generan la planeación una vez, pero la puesta en marcha es totalmente cíclica y retroalimentativa, porque al administrar los requerimientos se puede ver algún cambio y se debe retomar actividades por las cuales el plan ya pasó. Por lo tanto, la frecuencia es indeterminada y se puede presentar en cualquier fase, incluso durante la implementación de dichos requerimientos.

Para la puesta en marcha de la planeación del proceso de ingeniería se especifican actividades cruciales que toman papel importante que cada uno de los requisitos se plantee de manera correcta conforme a la necesidad de negocio que Smart Closet concibe dentro de sus objetivos. En la *Figura 3. Diagrama proceso de ingeniería de requisitos* se detalla cada actividad pertinente en el proceso junto con que el flujo de seguimiento entre actividades y, asimismo, se detalla el actor que interviene en dicha actividad.

El evento inicial del proceso de ingeniería de requisitos inicia con el subproceso de levamiento de requerimientos el cual está especificado en la [Sección 9.2 Levantamiento](#), cuando este conjunto de tareas termine, se especifica cada requisito para que cumpla con los criterios de aceptación, luego se lleva a cabo el subproceso de validación de requisito para validar si se especificaron de manera correcta los requisitos que pasaron por la actividad anterior, dicho flujo se puede ver con mayor explicación en la [Sección 10. Proceso de verificación y validación](#).

Una vez que los requisitos se validan se procede con un subproceso de administración de requisitos en el cual se hace la trazabilidad de estos y se evalúa si es necesario algún cambio, estas actividades se detallan en la *Sección 10.1.2 Administración de requerimientos* del documento SPMP. Una vez que se realizan las tareas pertinentes se da por terminado el proceso de ingeniería de requisitos.



*Ilustración 3. Proceso de ingeniería de requisitos.*

## 9.2. Levantamiento

Las técnicas de levantamientos de requisitos en ambientes profesionales maduros suelen estar acompañado de un análisis profundo de la documentación del proyecto, como planes de negocio y actas de constitución. Herramientas como entrevistas, encuestas y mesas de trabajo son normalmente usadas pues son muy útiles para comprender el entorno de los usuarios, clientes y otros interesados con el fin de poder lograr mejores aproximaciones, sin embargo, estas opciones son claramente inviables en este proyecto dado que se cumplen roles de responsables y dueños del mismo. El desarrollo del SPMP le permitió al equipo de Software Plus un gran entendimiento del proyecto, la creación de mockups fue útil para identificar claramente las principales funcionalidades del proyecto y por consiguiente los requisitos asociados a éste. Así que, el proceso de levantamiento de requerimientos se efectuó principalmente con los integrantes y dueños del sistema.

En ese sentido Software Plus hizo el proceso de levantar requisitos usando solamente su conocimiento del proyecto, dejando abierta la posibilidad de agregar nuevos requisitos. Los principales encargados de plasmar los requisitos fueron el Arquitecto de Software y el director del grupo con aportes de los demás integrantes en las tareas de este proceso.

Se realizó una el levantamiento de requerimientos desde una etapa temprana del proyecto, pero a medida que éste fue evolucionando se siguieron las actividades posteriores para formalizar y pulir aquellos requisitos para que posteriormente se puedan consignar en la documentación y de esta manera sean útiles para

desarrollar las funcionalidades del aplicativo web. Sí bien este proceso se realizó una vez, siempre se hace cada dos semanas una verificación con el fin de evaluar si se encuentran más requisitos pertinentes dentro del proyecto y se pueda madurar la estructura del sistema.

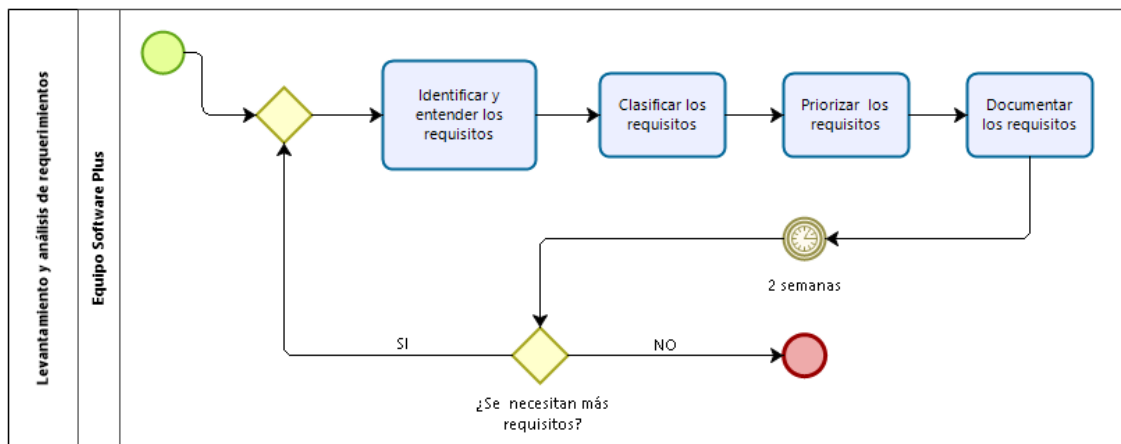


Ilustración 4. Proceso de levantamiento de requisitos

### 9.3. Administración de requisitos

En esta subsección se realizará una matriz de trazabilidad de los requisitos que facilitará la ubicación e impacto de éstos en el proyecto. Dicha matriz contará con la siguiente información:

- **ID:** Diferenciador de cada requisito. Se compone de la siguiente estructura: R (Categoría) (Tipo) (número de requisito del tipo)
- **Requisito:** Descripción corta del requisito
- **Estado:** Validado / Especificado / Verificado / Implementado
- **Versión:** Ultima versión del requisito
- **Caso de uso:** Identificador de casos de uso asociados.
- **Aspecto de aceptación:** Descripción del criterio con el cual se acepta el requisito.
- **Comentarios adicionales:** Texto libre y opcional para comentar la trazabilidad del requisito.

Estas especificaciones se encuentran en el [Anexo 3. Matriz de trazabilidad](#).

## 10. Proceso de verificación y validación

En esta sección se realizará la validación y verificación de los requisitos funcionales previamente identificados que componen el software Smart Closet. Esto se realiza teniendo en cuenta los siguientes atributos:

- **Atómico:** Único propósito, funcionalidad indivisible.
- **Correcto:** Inclusión exclusiva de los que fueron analizados previamente.
- **No ambiguo:** Única interpretación sin importar quién los lea.
- **Completo:** Contención de toda la información necesaria para entenderlos.
- **Consistentes:** Coherencia entre ellos. No deben contradecirse.
- **Importancia:** Clasificación según orden de importancia en el sistema.
- **Verificable:** Existe un criterio para medir si se cumplen los objetivos.
- **Modificable:** Baja cohesión para hacer modificaciones de manera sencilla.
- **Trazable:** Fácilmente mapeable. Origen claro.
- **Asociados a versión:** Orden de acuerdo a su evolución.
- **No redundante:** Sin repetición.
- **Preciso:** No necesitan de explicaciones adicionales.

Para la validación se realizó en una hoja de cálculo una tabla donde cada requerimiento está asociado con cada uno de los atributos anteriormente mencionados. Si el requerimiento cumple con cierto atributo se ingresa 'sí' y de esta manera cada requerimiento por cada uno de los 12 atributos. Al final se evalúa cada requisito y solo aquellos que cumplan con los 12 'sí' serán aprobados. Esta hoja se encuentra en el [Anexo 4. Validación de requerimientos](#). La *Figura 5. Diagrama de flujo para validar un requisito* muestra de forma gráfica este proceso.

Después de realizado este proceso se definieron un total de 18 requisitos que cumplen con los 12 criterios de validación especificados.

En la parte de verificación se planteó el proceso que se va a realizar para verificar cada uno de los requisitos identificados, tanto funcionales como no funcionales, donde cada uno de ellos es sometido a diferentes pruebas dependiendo de su naturalidad, la salida obtenida es evaluada teniendo en cuenta tres aspectos: Si la salida es correcta y según lo esperado, si el entorno quedo coherente según la salida o si aquellos aspectos que no se tenían que modificar quedaron iguales y si la cantidad de recursos utilizados corresponden de manera acorde con el requisito (aspectos como tiempo, memoria utilizada, entre otros). Si lo anteriormente mencionado se cumple el requisito cambia de estado ha verificado, si no, se replantea el requisito o se modifica según la necesidad y se vuelve a verificar. La *Figura 6. Diagrama de flujo para verificar un requisito* muestra de forma gráfica este proceso.



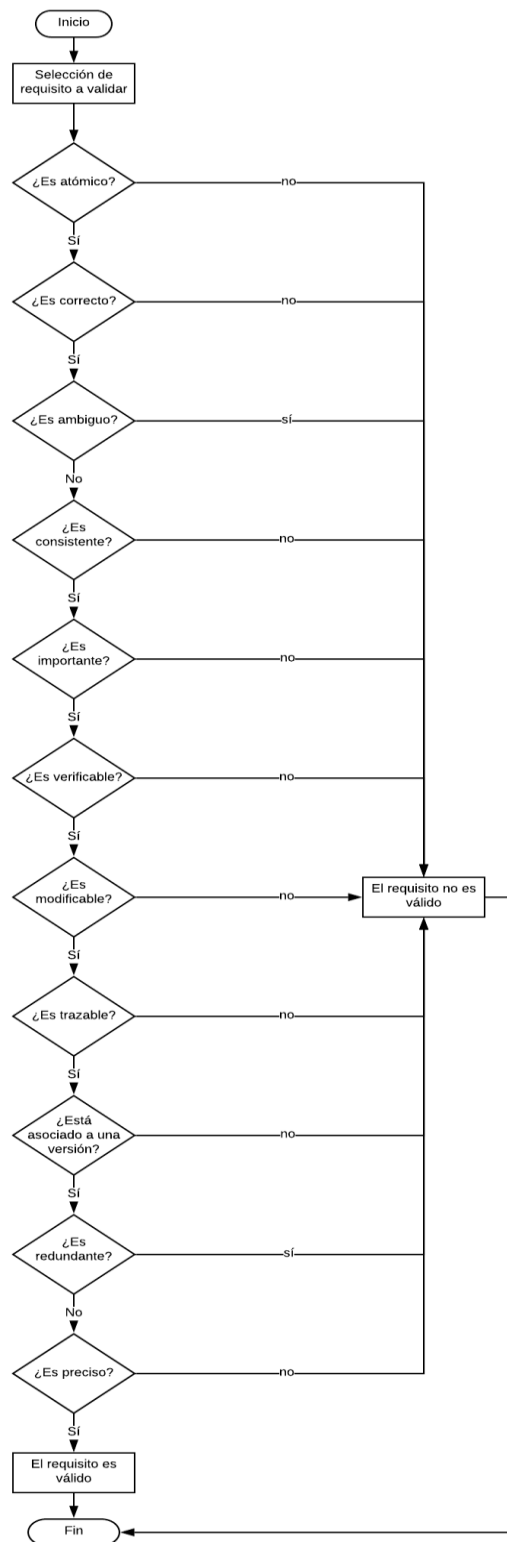


Ilustración 5. Diagrama de flujo para validar un requisito

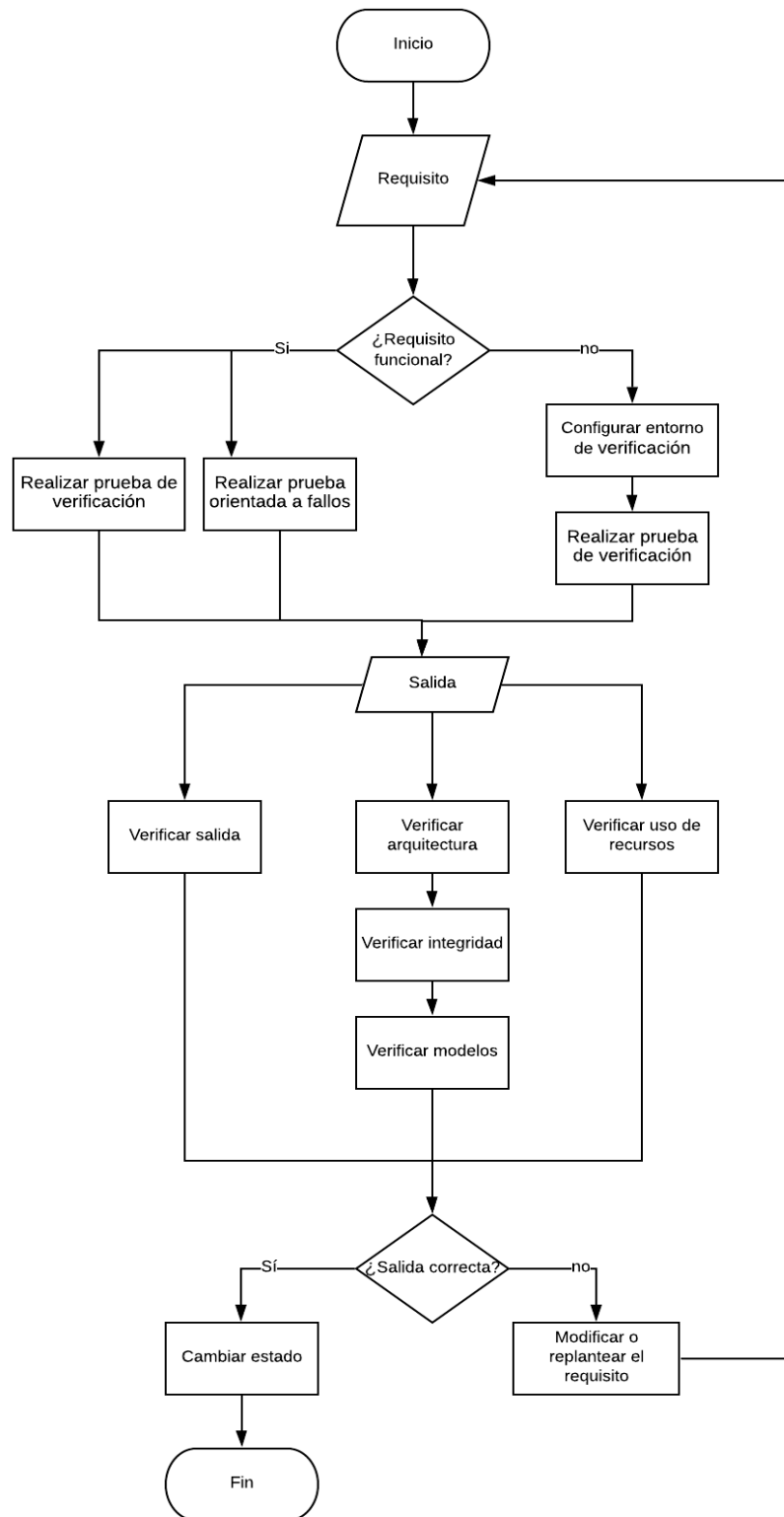


Ilustración 6. Diagrama de flujo para verificar un requisito.

## 11. Referencias

- [1]. PostgreSQL. 2020. Documentation: 9.3: Pg\_Dump. [online] Available at: <<https://www.postgresql.org/docs/9.3/app-pgdump.html>> [Accessed 8 April 2020].
- [2]. Sommerville, I., & Sawyer, P. (1997). *Requirements engineering: a good practice guide*. John Wiley & Sons, Inc
- [3]. Spring Framework. 2020. Web Applications. [online] Available at: <<https://spring.io/web-applications>> [Accessed 6 April 2020].
- [4]. H2database.com. 2020. H2 Database Engine. [online] Available at: <<https://www.h2database.com/html/main.html>> [Accessed 6 April 2020].
- [5]. H2database.com. 2020. Advanced. [online] Available at: <<http://www.h2database.com/html/advanced.html>> [Accessed 6 April 2020].