**2014**

Stephanie Buitrago

24/03/2014



SRS

**TABLA DE CONTENIDO**

LISTA DE FIGURAS 2

LISTA DE TABLAS 3

1. INTRODUCCIÓN 4

1.1 ALCANCE 4

1.2 OBJETIVOS 4

1.2.1 Objetivo General 4

1.2.2 Objetivos Específicos 4

1.3 REFERENCIAS 4

2. Descripción Global 4

2.1 Perspectiva del Producto 4

2.1.1 Interfaces con el sistema 5

2.1.2 Interfaces con el Usuario 5

2.1.3 Interfaces del Hardware 5

2.1.4 Interfaces de Comunicación 5

2.2 Funciones de Producto 5

2.3 Características del Usuario 6

2.4 Restricciones del Usuario 6

2.5 MODELO DEL DOMINIO 7

2.5.1 Diagrama Modelo del dominio 7

2.6 Suposiciones 7

2.7 Distribución de requerimientos 7

3. PLAN DE REQUERIMIENTOS 7

3.1 IDENTIFICACIÓN 7

3.2 ESPECIFICACIÓN 8

3.3 PRIORIZACIÓN 9

3.4 VALIDACIÓN 9

4. REQUERIMIENTOS ESPECIFICOS 10

4.1 CARACTERÍSTICAS DEL PRODUCTO DE SOFTWARE 10

4.2 ATRIBUTOS DEL SISTEMA DE SOFTWARE (NO FUNCIONALES) 11

5. ANEXOS 11

5.1 Referencias 11

5.2 Requerimientos 11

5.3 Casos de uso 11

# 

# LISTA DE FIGURAS

Ilustración 1: Clasificación de Requerimientos 10

Ilustración 2: Interfaces con el hardware 14

Ilustración 3: Requerimientos – Interfaces con el Hardware 15

Ilustración 4: Interfaces con el Software – Sistema Operativo 16

LISTA DE TABLAS

Tabla 1: Historial cambios 3

Tabla 2: Interfaces con el usuario 14

Tabla 3: Restricciones 17

1. INTRODUCCIÓN

El propósito de este documento es identificar, especificar y priorizar los requerimientos del prototipo de aplicación que se desarrollara como trabajo de grado de la alumna Stephanie Buitrago, para comprender la estructura funcional y no funcional del diseño que presenta la aplicación.

Un SRS reduce al mínimo el tiempo y el esfuerzo requerido por los desarrolladores para alcanzar los objetivos deseados y también minimiza el coste de desarrollo. Un buen SRS define como una aplicación va a interactuar con el hardware del sistema, otros programas y los usuarios humanos en una amplia variedad de situaciones del mundo real. [[104]](#_REFERENCIAS)

## ALCANCE

La identificación y especificación de los requerimientos funcionales y no funcionales del sistema, para estipular su funcionalidad.

## OBJETIVOS

A continuación se detallarán los objetivos del actual documento.

### Objetivo General

#### Determinar las características que el sistema tendrá. La especificación de los requerimientos se realizará a lo largo de la implementación del presente sistema de información.

### Objetivos Específicos

* Verificar y validar los requerimientos recolectados que cumplan con el desarrollo de la aplicación, durante el transcurso del proyecto.
* Dar inicio con la construcción del prototipo, a partir de los requerimientos recolectados durante este hito e implementando los casos de uso descritos con este documento.

## REFERENCIAS

[Ver Anexo Referencias](#_Referencias_77)

# Descripción Global

Este documento se encarga de brindar una descripción completa sobre la funcionalidad del software y la expectativa sobre el producto por parte del desarrollador y de los clientes. Igualmente, este es un documento de soporte para los programadores, facilitándoles la implementación del producto en los periodos académicos que se requieran.

## Perspectiva del Producto

Este producto será una aplicación independiente, que responderá con los requisitos propuestos y tendrá un diseño modular para gestionar el proceso de implementación.

### Interfaces con el sistema

Debido a que la aplicación no interactuará con ningún otro sistema de software, no es necesario especificar su dependencia con otras interfaces.

### Interfaces con el Usuario

Representa todos los factores que intervienen con el usuario para interactuar con la aplicación, permitiendo facilitar al usuario acceder a ésta.

* Teclado: Dispositivo de entrada utilizado para navegar e interactuar con los diferentes componentes del producto de software.
* Mouse: Utilizado para navegar y controlar las diferentes piezas de la aplicación a través del tablero de este producto de software.
* Pantalla: Recurso utilizado para la interacción visual del usuario con el producto de software. En el caso en el que se acceda a la aplicación a través de un dispositivo móvil, este recurso aplica como dispositivo de entrada y de control además de la interacción visual.
* Interfaz Gráfica de Usuario: Esta interfaz provee una ambiente agradable que interactúa con el usuario final.

### Interfaces del Hardware

Las interfaces del hardware son las asociadas a nivel de los dispositivos físicos, utilizados para ingresar, procesar y entregar los datos. Para el correcto funcionamiento de esta aplicación web, es imperativo que cada dispositivo cuente con una Tarjeta de Red que permita establecer la conexión entre el servidor y el navegador de Internet de cada terminal del cliente. Además, es indispensable el uso de dispositivos de salida y entrada como mouse, teclado y pantalla, en el caso de los dispositivos móviles como celulares y tabletas se deberá contar con teclado y pantalla táctil.

### Interfaces de Comunicación

Es necesario disponer del protocolo TCP/IP orientado a conexión, entre la aplicación y el servidor central que proveerá el flujo constante de datos. Las aplicaciones indispensables, tanto el cliente como el servidor, harán uso del puerto destinado por el entorno de programación y de acuerdo al protocolo especificado.

## Funciones de Producto

Durante el transcurso de la realización de este documento, se ha estado especificando todas las funciones del producto en la documentación de los casos de uso (Ver anexo [Casos de Uso.docx](#_Casos_de_uso_1)).

A continuación se mostrarán las principales funciones, tal como fueron definidas en los requerimientos funcionales (Ver anexo [Requerimientos](#_Requerimientos_6)):

* Se manejará un modelo arquitectónico MVC (Modelo Vista Controlador).
* El servidor debe tener la capacidad de recibir varias solicitudes de manera concurrente.
* El sistema deberá ser responsivo de acuerdo al dispositivo mediante el cual se acceda.
* El usuario podrá crear su perfil.
* El usuario podrá compartir información de su gusto en el contexto de la moda.
* El usuario podra ver en la aplicación información relevante del contexto de la moda.

## Características del Usuario

Las características del usuario se encuentran definidas dentro del Documento Casos de Uso (Ver anexo [Casos de Uso](#_Casos_de_uso_2)).

## Restricciones del Usuario

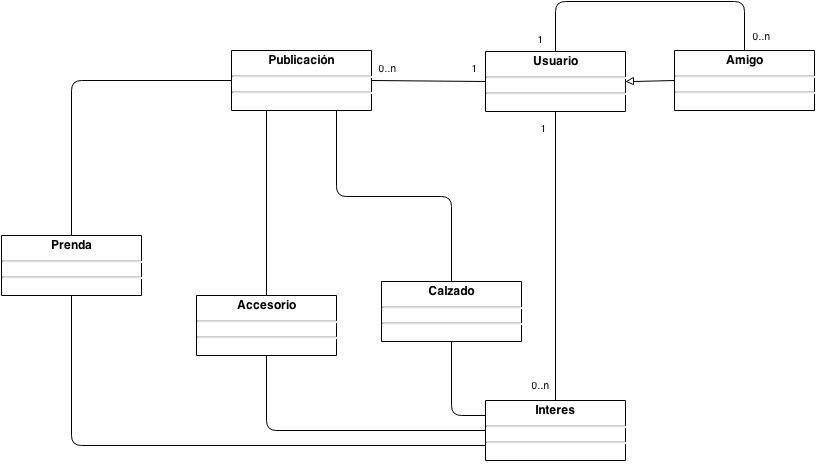
|  |  |
| --- | --- |
| **Tipo de Restricción** | **Restricciones** |
| **Arquitectura** | El producto tendrá una arquitectura Modelo, Vista Controlador, según lo definido en los requerimientos no funcionales (Ver anexo [Requerimientos](#_Requerimientos_7)). |
| **Cliente** | El prototipo funcional debe ser entregado en las fechas especificadas por la facultad de ingeniería.  La aplicación debe funcionar correctamente tanto en dispositivos moviles como en dipositivos de escritorio. |
| **Interfaz de usuario** | La interfaz deberá ser sencilla y fácil de utilizar para el usuario.  El idioma de la aplicación se visualizará en Español (Colombia). |
| **Lenguaje de Programación** | La aplicación será desarrollada en HTML 5, CSS, Ruby. |
| **Legales** | Todo contenido obtenido de otras fuentes siempre será referenciado. |
| **Persistencia** | Solo se manejarán bases de datos en MYSQL, porque es necesario guardar un gran volumen de registros en tablas relacionales. |
| **Sistema operativo** | El sistema debe contar con una conexión a internet.  El sistema debe contar con un navegador. |

# 

## MODELO DEL DOMINIO

A continuación se presenta el modelo obtenido a partir del análisis que se realizó del dominio de la moda.

### Diagrama Modelo del dominio



#### Usuario

Es la representación del usuario que utiliza la aplicación

* Puede tener uno o más amigos
* Puede tener uno o intereses.
* Puede realizar una o más publicaciones.

#### Interés

Es la representación de un gusto o preferencia de un usuario.

* Un interés pertenece a un usuario
* Este se relaciona con cada una de las categorías, prenda, accesorio y calzado.

#### Publicación

Es la representación de una publicación que realiza un usuario.

* Una publicación pertenece a un usuario
* Este se relaciona con cada una de las categorías, prenda, accesorio y calzado.

#### Prenda

Es la representación de la categoría de prenda, está tiene como atributos: tipo, material, color, talla y clima

#### Calzado

Es la representación de la categoría de calzado, está tiene como atributos: tipo, material, color, talla y clima

#### Accesorio

Es la representación de la categoría de Accesorio, está tiene como atributos: tipo, material, color, talla y clima

## Suposiciones

Dentro de las suposiciones podemos encontrar:

* Se deberá cumplir con las entregas establecidas (ver Sección 1.1.3 del documento SPMP).
* El sistema no depende de la interacción de otras interfaces, por lo cual no se tienen dependencias.

## Distribución de requerimientos

Para realizar la distribución de requerimientos, se identificaron las necesidades de negocio, luego se refinaron los casos de uso, se identificaron los requerimientos, además estos fueron especificados y después se distribuyeron entre funcionales y no funcionales. (Ver Sección 3.2 [***Especificiación de Requerimientos***](#_ESPECIFICACIÓN)).

# PLAN DE REQUERIMIENTOS

## IDENTIFICACIÓN

La identificación de los requerimientos es necesaria para el desarrollo del software, se debe tener en cuenta el contexto en el que se pretende desarrollar la aplicación, la población objetivo, y la meta que se quiere conseguir con la implementación del proyecto. Además se deberá tener en cuenta el alcance definido, esto debido al tiempo que toma el desarrollo vs el tiempo dado para desarrollar.

## ESPECIFICACIÓN

La especificación de los requerimientos se realiza de la siguiente manera, según los criterios definidos, basándose en la plantilla de “Requerimientos Templates - Contrux[[200]](#_Referencias_7)” con algunas modificaciones pertinentes que se adapten al desarrollo del software.

Esta plantilla contiene los parámetros necesarios para llevar a cabo una adecuada documentación de los requerimientos, a continuación se muestra cada uno de los parámetros con su respectiva definición:

* **ID Requerimiento:** Identificador único de cada requerimiento.
* **Versión:** Representa los cambios que se efectúan en el requerimiento.
* **Autor:** Nombre de la persona encargada de especificar cada requerimiento y la gestión del mismo.
* **Prioridad:** Es la importancia de cada requerimiento dentro del sistema (Ver Sección 3.3 [Priorización requerimientos](#_PRIORIZACIÓN_5)).
* **Fecha de Creación:** Creación de cada requerimiento en formato DD/MM/AAAA.
* **Fecha de Modificación:** Modificación de cada requerimiento en formato DD/MM/AAAA.
* **Descripción:** Definición de cada requerimiento.
* **Estado:** Define el seguimiento de cada requerimiento, para ello se establecieron los siguientes parámetros de estado que fueron tomados de “Software Requirements” [[2]](#_Referencias_6) y aplicadas al proyecto con sus respectivos porcentajes:

**Propuesto:** El requerimiento ha sido solicitado por una fuente autorizada (Arquitecto o Director de Requerimientos). *Valor Porcentual: 0% – 5%*

**Aprobado:** El requerimiento ha sido analizado, documentado y su impacto en el proyecto se ha estimado. El grupo de arquitectura ha decidido incorporar el requerimiento y el grupo de desarrollo de software se ha comprometido a implementarlo. *Valor Porcentual: 5% – 15%*

**Implementado:** El código que implementa el requerimiento, ha sido diseñado, programado y probado unitariamente. Los requerimientos han sido asociados al diseño pertinente y al código fuente correspondiente. *Valor Porcentual: 15% – 80%*

**Verificado:** El funcionamiento correcto del requerimiento implementado ha sido conformado en el producto integrado, ha sido asociado a los pertinentes casos de pruebas. El requerimiento ahora es considerado completo. *Valor Porcentual: 80% – 100%*

**Eliminado:** Un requerimiento aprobado previamente ha sido eliminado. Incluye una explicación de por qué y quién tomo la decisión de eliminarlo.

**Rechazado:** El requerimiento fue propuesto, pero no está previsto para la implementación en una próxima entrega, Incluye una explicación de por qué y quién decidió rechazarlo.

* **Requerimiento(s) Asociado(s):** Son aquellos requerimientos que se complementan.

En el siguiente documento se pueden encontrar cada uno de los requerimientos Funcionales y No Funcionales, considerados pertinentes para el desarrollo de la aplicación. ([Ver anexo Requerimientos](#_Requerimientos_8))

## PRIORIZACIÓN

Se definirán cuáles son los requerimientos más importantes del proyecto para priorizar su implementación, además conseguir que el prototipo presentado sea acorde a la propuesta y supla las necesidades encontradas.

Para esto se definirán y utilizarán unas métricas que permitirán la priorización de los requerimientos con lo que se pretende lograr una mejor implementación y el seguimiento del progreso del proyecto.

Definición de métricas

Tiempo de ejecución.

Esta variable fue definida en la estimación del proyecto dada en horas y sirve para referirnos al tiempo para de desarrollar un requerimiento, ésta no solo tiene en cuenta los tiempo de desarrollo, también los tiempos necesarios para el entrenamiento, de desplazamiento, y de instalación de infraestructura.

Casos de uso con alta cohesión

Esta métrica nos indica que tan correlacionado está el caso de uso asociado con los demás, identificando entonces los requerimientos que tengan este caso de uso, esto para definir que requerimientos son más importantes que otros y cuales nos generan más desarrollo en el sentido de abarcar más en su implementación.

Dificultad por parte del desarrollador

Se identificarán cuáles son los requerimientos de difícil implementación para el desarrollador, también se identifica si ese requerimiento necesita una investigación previa para su desarrollo.

## VALIDACIÓN

La validación de los requerimientos debe mostrar que los requerimientos están definiendo de manera correcta el sistema que el cliente necesita. La validación de los requerimientos comienza desde la primera versión del documento de los requerimientos para cumplir con un proceso correcto de aceptación [[201]](#_Referencias_8).

En esta etapa, es indispensable la corrección de los requerimientos para evitar costos adicionales durante el desarrollo o la implementación del producto. Se realizará esta labor siguiendo los siguientes pasos:

* **Verificación de Validez**: Cuando se trabaja con los requerimientos, es común analizar durante la construcción que estos enunciados carecen especificación sobre la funcionalidad. Para evitar retrasos y discusiones con los desarrolladores, se redefinen estos requisitos del sistema.
* **Verificación de Consistencia**: Después de la recolección de requerimientos, es habitual que en cada requisito se encuentren contradicciones. Lo cual significa que se deben analizar estas aseveraciones que especifican contrariamente lo que otros requisitos afirman, y se decide si se debe replantearlo o rechazarlo.
* **Verificación de integridad**: Se encargará de analizar si se ha pasado por alto algún requisito, realizando una lectura adicional a los documentos relacionados con la implementación de la arquitectura y comparando con el contenido del documento de requerimientos.
* **Verificación de Trazabilidad**: Se debe asegurar que estos requerimientos logren ser implementados completamente en el producto por entregar, teniendo en cuenta restricciones de planeación.
* **Verificabilidad**: Se procede a la redacción de los requerimientos con el fin de confrontar las expectativas del cliente frente a las aseveraciones recolectadas.

Todas estas exigencias, anteriormente planteadas, facilitarán el control de buenos requerimientos cumpliendo con las siguientes características:

* **No ambiguo**: Debe ser claro, alcanzable, preciso y tener una única interpretación posible.
* **Correcto**: Debe estar bien redactado.
* **Completo**: deben contener en sí mismos toda la información necesaria, y no remitir a otras fuentes externas que los expliquen con más detalle
* **Consistente**: Ningún requisito debe entrar en conflicto con otro requisito diferente, ni con parte de otro. Asimismo, el lenguaje empleado entre los distintos requisitos debe ser consistente también.
* **Escrito en forma de “Debe”**: Debe iniciar en forma “Debe”.
* **No repetido**: puede existir uno redactado de otra forma que tenga el mismo objetivo.
* **Verificable**: poder verificar con absoluta certeza, si el requisito fue satisfecho o no. Esta verificación puede lograrse mediante inspección, análisis, demostración o testeo.
* **Escrito en lenguaje natural**: Debe ser claro para los que necesitan hacer uso de los requerimientos.
* **Necesario**: Lo que pida un requisito debe ser necesario para el producto.
* **Puede seguirse**: Deben tener artefactos asociados a lo largo del ciclo de vida del producto.

# REQUERIMIENTOS ESPECIFICOS

Esta sección del documento contiene todos los requerimientos definidos para el proyecto.

## CARACTERÍSTICAS DEL PRODUCTO DE SOFTWARE

Los requerimientos relacionados a las características del sistema son detallados y especificados en el siguiente anexo ([ver Anexo Requerimientos](#_Requerimientos_10)).

## ATRIBUTOS DEL SISTEMA DE SOFTWARE (NO FUNCIONALES)

Los requerimientos relacionados a los atributos del sistema, son detallados y especificados en el siguiente anexo en la pestaña de Requerimientos no funcionales ([ver Anexo Requerimientos](#_Requerimientos_11)).

Para la obtención de requerimientos no funcionales, se especificó estos requisitos, tratándose de conexión, soporte a fallos, modo de persistencia y requisitos mínimos de los equipos donde se probará el funcionamiento del sistema.

1. ANEXOS

## Referencias



## Requerimientos



## Casos de uso

