



# Especificación de Requerimientos

## Descripción del Diseño

### PetLost

## Localización de mascotas perdidas

Apellidos, Nombres	Correo electrónico	Rol
<b>Vallejo Salazar, Jorge Antonio</b>	jorgevallejo2990@gmail.com	Gestor de Proyecto
<b>Castro Valencia, Cristian</b>	cristiancastro9909@gmail.com	Gestor Base de datos
<b>Montero Corzo, Jesús David</b>	jesusda71.2@gmail.com	Desarrollador Back-end
<b>Useche Perdomo, Joshua Daniel</b>	joshua_dup@hotmail.com	Desarrollador Front-end
<b>Ardila Madero, Leonardo José</b>	lejoarma@hotmail.com	Tester

Fecha de presentación: 20/09/2022

## Contenido

<b>1</b>	<b>INTRODUCCIÓN.....</b>	<b>3</b>
1.1	OBJETIVO .....	3
1.1.1	<i>Público Objetivo:</i> .....	3
1.1.2	<i>Impacto Esperado:</i> .....	3
1.2	ALCANCE O ÁMBITO DEL SISTEMA .....	3
1.3	DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS .....	6
1.3.1	<i>Definiciones</i> .....	6
1.3.2	<i>Acrónimos</i> .....	6
1.3.3	<i>Abreviaturas</i> .....	6
1.3.4	<i>Referencias</i> .....	6
1.4	PERSPECTIVA GENERAL DEL DOCUMENTO.....	6
<b>2</b>	<b>DESCRIPCIÓN GENERAL DE LA APLICACIÓN .....</b>	<b>7</b>
2.1	PERSPECTIVA DE LA APLICACIÓN.....	7
2.2	FUNCIONES DE LA APLICACIÓN.....	7
2.3	CARACTERÍSTICAS DE LOS USUARIOS .....	7
2.4	RESTRICCIONES.....	8
2.5	SUPOSICIONES Y DEPENDENCIAS .....	8
2.6	REQUERIMIENTOS DIFERIDOS .....	8
<b>3</b>	<b>REQUERIMIENTOS ESPECÍFICOS.....</b>	<b>8</b>
3.1	REQUERIMIENTOS .....	8
3.1.1	<i>Product Backlog</i> .....	8
3.1.2	<i>Ciclo de Sprints del proyecto</i> .....	9
3.1.3	<i>Sprint Backlog</i> .....	10
3.1.4	<i>Historias de usuario (Tareas y Subtareas)</i> .....	10
3.1.5	<i>Mecánica de organización del grupo. (Reuniones, evidencias/artefactos)</i> .....	10
3.2	MODELO DE REQUERIMIENTOS .....	10
3.2.1	<i>Modelo de Casos de Uso</i> .....	10
<b>4</b>	<b>DESCRIPCIÓN DEL DISEÑO .....</b>	<b>12</b>
4.1	INTERFAZ GRÁFICA (MOCKUPS).....	12
4.2	PALETA DE COLORES USADA. ....	13
<b>5</b>	<b>GESTIÓN DE LA CONFIGURACIÓN .....</b>	<b>13</b>
<b>6</b>	<b>PRUEBAS .....</b>	<b>13</b>
6.1	DESCRIPCIÓN DE PRUEBAS UNITARIAS .....	13
6.2	DESCRIPCIÓN DE PRUEBAS DE ACEPTACIÓN .....	14
<b>7</b>	<b>GLOSARIO .....</b>	<b>14</b>
<b>8</b>	<b>ANEXO(S) .....</b>	<b>14</b>

# 1 INTRODUCCIÓN

Los animales nos otorgan continuamente un amor incondicional y por esa razón poco después de que lleguen a nuestras vidas se transforman en un integrante más del núcleo familiar. Quien ha pasado por la pérdida de una mascota sabe el dolor que implica este caso. No es simple aceptar que nuestro estimado amigo por el momento se encuentra perdido, extraviado, o simplemente no volverá más, como tampoco volverán los infinitos instantes de alegría y cariño que nos regalaba todos los días.

Por lo tanto, este recurso está diseñado para ayudar a encontrar una mascota perdida, así como proporcionar herramientas de información y/o contacto entre dueños y personas que encuentran mascotas, formularios para informar una pérdida con información detallada del evento de pérdida. Además, aquellos que encuentran a una mascota perdida deben encontrar formas confiables de devolver el animal a casa de manera segura.

## 1.1 Objetivo

Facilitar la búsqueda, localización y recuperación de mascotas perdidas por medio del uso de un aplicativo web.

### 1.1.1 Público Objetivo:

Familias y dueños de mascotas.

### 1.1.2 Impacto Esperado:

Aumento en las cifras de recuperación de mascotas extraviadas.

## 1.2 Alcance o Ámbito del Sistema

### a) PetLost

### b) El sistema deberá permitir el registro del usuario

El sistema deberá permitir el logueo a la plataforma

El sistema deberá permitir consultar mascotas perdidas

El sistema deberá permitir crear una mascota perdida

Plus:

El sistema deberá permitir actualizar una mascota existente

El sistema deberá permitir actualizar información del usuario

El sistema deberá permitir eliminar una mascota ya encontrada

### c) Módulo de Registro de usuario:

- Contará con:
- Un campo para diligenciar:

- Nombre
  - Apellido
  - Teléfono
  - Correo
  - Validador de campos
  - Botón de registro
- d) Módulo de inicio de sesión:
- Correo
  - Contraseña, visualizar contraseña
  - Olvidar Contraseña Plus
  - Botón inicio de sesión
- e) Modulo Perfil Usuario:
- Nombre
  - Email
  - Ciudad/Municipio
  - Dirección
  - Teléfono
  - Solo se vería la información registrada.
- f) Módulo para visualizar en forma de lista o mosaico de las (Se pueden poner filtros) publicaciones de mascotas perdidas.
- Solo se mostrarán anuncios con imagen de Perro o Gato
  - Se mostrará tarjeta con el nombre de la mascota, y su respectiva información, con nombre del dueño subrayado en la parte inferior derecha de la tarjeta.
  - Poder ordenar perdidas recientes Plus
  - Poder ordenar perdidas más antiguas Plus
  - Poder ordenar por ciudades Plus
  - Poder mostrar fecha de perdida en la tarjeta Plus
- g) Módulo para publicar anuncio de mascota perdida. (Se da toda la información)
- Nombre de la mascota
  - Descripción detallada realzando las características únicas, como: Color, Raza y manchas específicas.
  - Fecha de perdida
  - Lugar donde se perdió
  - Apartado de Cargar imagen Max 3 fotos por mascota perdida, idealmente recientes.
  - Botón cargar imagen

## h) Módulo de información de mascotas perdidas

- Nombre de la mascota
- Descripción de la pérdida
- Datos de la mascota, si es perro o gato
- Fecha de la pérdida
- Lugar de la pérdida
- Datos de contacto del dueño de la mascota como: Nombre, Teléfono y Correo.
- Botón de contactar. Plus

## i) ? Módulo para confirmar que se ha visto la mascota

- Lugar
- Día
- Detalles
- Teléfono de contacto de la persona que lo vio.
- Plus

## j) Módulo de actualización de datos desde el perfil del usuario

- Actualización de datos
- Nuevo teléfono
- Nueva dirección
- Nuevo correo
- Plus

## k) Editar publicación de mascotas perdidas

- Plus

## l) Poder eliminar publicación de mascota perdida

- Plus

## m) Eliminar mascotas encontradas de la DB

- Plus

Para ingresar a la aplicación se deberá colocar el usuario y contraseña de la persona, si no tiene uno deberá registrarse primero. La información que se pedirá de la persona será: Nombre, Apellido, Nombre de Usuario, Correo Electrónico, Teléfono, Cedula. Al realizar el login, se mostrará como pantalla principal una lista de mascotas perdidas en orden de fecha de publicación desde la más reciente hasta la más antigua. Desde la pantalla principal, se tendrá un botón para publicar una mascota perdida. La información de la mascota se almacenará en la base de datos y contendrá la siguiente información: Nombre, Raza, Color, descripción. Desde el perfil del usuario, puede modificar sus datos personales y se verán sus publicaciones, dónde podrá eliminar o modificar la publicación.



[Este apartado debe:

- a) identificar por nombre la aplicación web que se producirá;
- b) explicar lo que hará la aplicación y, si es necesario, lo que no hará;
- c) describir el uso de la aplicación, incluyendo beneficios relevantes, objetivos y metas.]

### **1.3 Definiciones, Acrónimos y Abreviaturas**

[Este apartado debe proporcionar las definiciones de términos, siglas y abreviaturas necesarios para interpretar adecuadamente el documento. Esta información puede proporcionarse por referencia a uno o más anexos de este documento o por referencia a otros documentos.]

#### 1.3.1 Definiciones

#### 1.3.2 Acrónimos

#### 1.3.3 Abreviaturas

#### 1.3.4 Referencias

[Este apartado debe:

- a) proporcionar una lista completa de todos los documentos a los que se haga referencia;
- b) identificar cada documento por título, número de informe (si es procedente), fecha y organización que lo publicó.]

### **1.4 Perspectiva General del Documento**

[Este apartado debe:

- a) describir lo que contiene el resto del documento;
- b) explicar cómo está organizado el documento.]

## 2 DESCRIPCIÓN GENERAL DE LA APLICACIÓN

[Esta sección del documento debe describir los factores generales que afectan a la aplicación y sus requerimientos. Esta sección no establece requerimientos específicos, los que se describen en detalle en la sección 3, sino que proporciona un contexto para dichos requerimientos.]

Generalmente, esta sección consiste en seis apartados, que son los que siguen.]

### 2.1 Perspectiva de la Aplicación

[Este apartado debe poner la aplicación en perspectiva con otros productos relacionados. Si el producto es totalmente autónomo e independiente, se debe indicar aquí. Si se define una aplicación que es un componente de un sistema más grande, como ocurre con frecuencia, entonces en este inciso se debe relacionar los requerimientos del sistema más grande a la funcionalidad de la aplicación y debe identificar las interfaces entre ese sistema y la aplicación. Puede ser útil un diagrama de bloques que muestre los principales componentes del sistema más grande, las interconexiones y las interfaces externas.]

Si son necesarios más detalles, recurrir al IEEE Std-830-1998.]

### 2.2 Funciones de la Aplicación

[Este apartado debe proporcionar un resumen de las principales funciones que ejecutará la aplicación, sin indicar la gran cantidad de detalles que pueda requerir cada una de esas funciones.]

A veces el resumen de funciones que se necesita para este apartado puede tomarse directamente de una especificación de nivel superior (si existe) que asigna funciones particulares para la aplicación. Téngase en cuenta que por motivos de claridad

a) Las funciones deben organizarse de una manera que haga que la lista de funciones sea comprensible para el cliente o para cualquier otra persona que lea el documento por primera vez.

b) Pueden utilizarse formas textuales o gráficas para mostrar las diferentes funciones y sus relaciones. Con un esquema tal, no se pretende mostrar el diseño de un producto, sino simplemente las relaciones lógicas entre las funciones.]

### 2.3 Características de los Usuarios

[Este apartado debe describir las características generales de los usuarios previstos de la aplicación incluyendo el nivel educativo, experiencia y conocimientos técnicos. No debe utilizarse para establecer requerimientos específicos.]



## 2.4 Restricciones

[Este apartado debe proporcionar una descripción general de cualquier otra cuestión que limite las opciones del desarrollador. Se podría incluir: a) políticas regulatorias; b) limitaciones de hardware (por ejemplo, requerimientos de sincronización de señales); c) interfaces a otras aplicaciones; d) operación paralela; e) funciones de auditoría; f) funciones de control; g) requerimientos de lenguajes de alto nivel; h) protocolos; i) requerimientos de confiabilidad; j) criticidad de la aplicación; k) consideraciones de seguridad.]

## 2.5 Suposiciones y Dependencias

[Este apartado debe enumerar cada uno de los factores que afectan los requerimientos establecidos. Estos factores no son restricciones de diseño sino que, por el contrario, cualquier cambio en ellos que podría afectar los requerimientos. Por ejemplo, una hipótesis puede ser que un sistema operativo específico estará disponible para la aplicación. Si, de hecho, el sistema operativo no está disponible, el documento tendrá que cambiar en consecuencia.]

## 2.6 Requerimientos Diferidos

[Este apartado debe identificar los requerimientos que pueden postergarse para versiones futuras de la aplicación.]

# 3 REQUERIMIENTOS ESPECÍFICOS

[Para el desarrollo de toda esta sección 3 utilizar como bibliografía de soporte:

(1) Schwinger, W.; Koch, N. "Modeling Web Applications", Chapter 3 en: Kappel, G.; Pröll, B.; Reich, S.; Retschitzegger, W. (Editors) *Web Engineering. The Discipline of Systematic Development of Web Applications*, John Wiley & Sons Ltd., 2006.

(2) Koch, N.; Knapp, A.; Zhang, G.; Baumeister, H. "UML-Based Web Engineering. An Approach Based on Standards", Chapter 7 en: Rossi, G.; Pastor, O.; Schwabe, D.; Olsina, L. (Editors) *Web Engineering: Modelling and Implementing Web Applications*, Springer-Verlag London Limited, 2008.]

## 3.1 Requerimientos

[Este apartado debe contener una lista de los requerimientos surgidos de las necesidades e ideas aportadas por los usuarios/clientes, desarrolladores y demás participantes en el proceso de desarrollo.]

### 3.1.1 Product Backlog

[Lista con todos los requerimientos iniciales del producto que se va a desarrollar que identifique las necesidades del producto para lograr su máxima utilidad. Asimismo, contiene

la descripción de las tareas y subtareas que se van a realizar para la ejecución de cada requisito, mismas que se organizarán en función de sus prioridades. Además, la pila de producto también indica una estimación del tiempo en la que cada tarea se va a desarrollar y el valor que cada una le da al producto.]

### 3.1.2 Ciclo de Sprints del proyecto

#### 3.1.2.1 Sprint 1

- Entrega en primera clase semana 4 (3 semanas para su desarrollo)
- Repositorio (GitHub) donde integrarán el trabajo.
- Gestión de configuración.
- Historias de usuario desarrolladas en el Sprint 1. (Trello)
- Presentación MVC
- Interfases gráficas de usuario
- Navegación entre interfases
- Informe de retrospectiva
- Historias de usuario a desarrollar en el sprint 2. (Trello)

#### 3.1.2.2 Sprint 2

- Entrega en primera clase semana 6 (2 semanas para su desarrollo)
- Presentación MVC
- Implementación de la lógica de negocio
- Pruebas unitarias de la lógica desarrollada
- Informe de retrospectiva
- Historias de usuario a desarrollar en el sprint 3. (Trello)

#### 3.1.2.3 Sprint 3

- Entrega en última clase semana 6 (1 semana para su desarrollo)
- Presentación MVC
- Aplicación con persistencia Relacional
- Pruebas unitarias de la lógica desarrollada
- Informe de retrospectiva
- Historias de usuario a desarrollar en el sprint 4. (Trello)

#### 3.1.2.4 Sprint 4

- Entrega en última clase semana 7 (1 semana para su desarrollo)
- Presentación MVC
- Despliegue Back-End en la nube
- Despliegue base de datos en local/nube
- Informe de retrospectiva

[Listar los sprints contemplados desde el proyecto para la generación de valor al cliente]

### 3.1.3 Sprint Backlog

[Lista de elementos seleccionados previamente del Product Backlog para ser desarrollados en el día a día en los diferentes Sprints del proyecto. Tras crear esta lista, el equipo del proyecto tendrá que identificar las funcionalidades y priorizar las que se entregarán en el Sprint.]

### 3.1.4 Historias de usuario (Tareas y Subtareas)

[Es una explicación general e informal de una función de software escrita desde la perspectiva del usuario final. Su propósito es articular cómo proporcionará una función de software valor al cliente. Asimismo, deben utilizar un lenguaje no técnico ya que supone la descripción de una función de software descrita por los usuarios finales].

### 3.1.5 Mecánica de organización del grupo. (Reuniones, evidencias/artefactos)

## 3.2 Modelo de Requerimientos

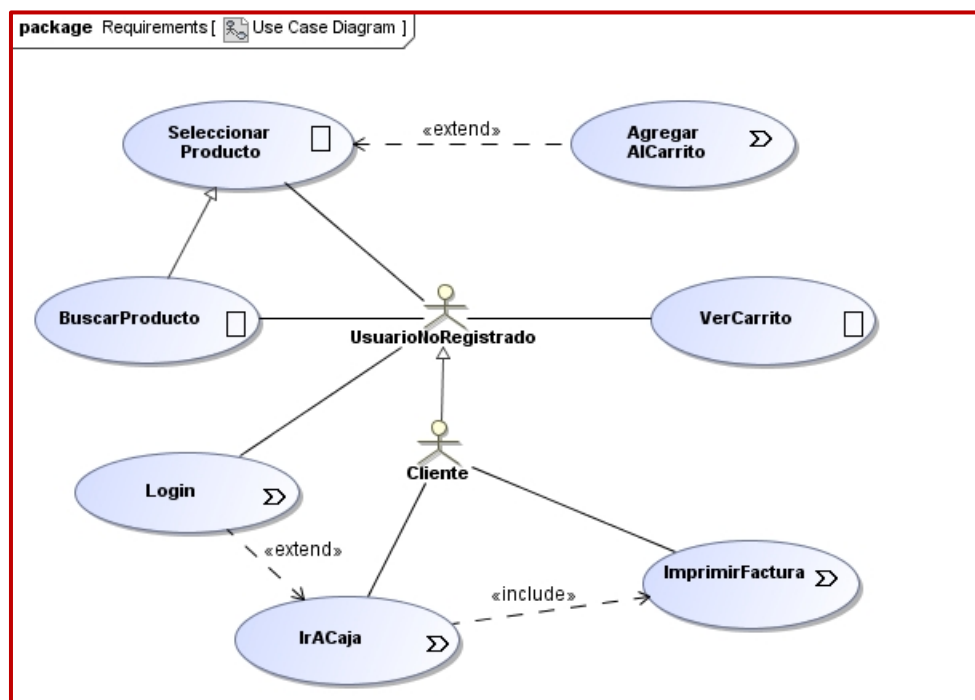
[El *modelo de requerimientos* se basa en dos modelos principales: el *modelo de casos de uso* y el *modelo del dominio*.]

### 3.2.1 Modelo de Casos de Uso

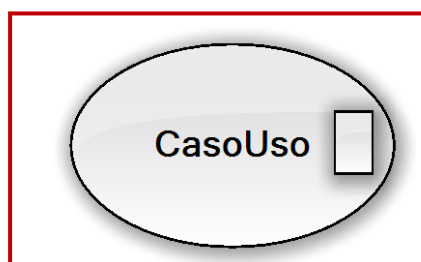
[La funcionalidad de una aplicación Web se modela como un conjunto de *casos de uso*, que describen los requerimientos de la aplicación desde las perspectivas de los *actores* (personas y otros sistemas).

Una particularidad de los requerimientos de una aplicación Web es la funcionalidad de navegación, que permite al usuario navegar por el hipertexto y encontrar nodos. El enfoque de UWE es crear un único modelo de casos de uso, que utiliza el estereotipo <<navigation>> para denotar la diferencia entre casos de uso funcionales y casos de uso específicos de hipertexto.

**OBSERVACIÓN:** Si la cantidad de casos de uso en un mismo diagrama es grande (una cantidad adecuada podría ser  $7 \pm 2$ ), sería conveniente aplicar el antiguo principio de "divide et impera". Es decir, dividir el diagrama de casos de uso en varias partes agrupando elementos que tengan algún tipo de coincidencia entre sí. El elemento de UML que se utiliza para estas situaciones es el *paquete*. Si todavía algún paquete tuviera demasiados casos de uso, entonces se podría subdividir ese paquete en varios paquetes, y dentro de estos estarían los casos de uso. Se debe asegurar que los paquetes sean cohesivos, es decir, que manejen información o funcionalidad relacionada. Luego, a partir del diagrama de paquetes se derivan los casos de uso; cada paquete implica un diagrama de casos de uso (o de paquetes).]



[Para el estereotipo <<navigation>>, en lugar de la etiqueta se puede usar el ícono (□).]



[A continuación, se describen los casos de uso utilizando tablas y diagramas.]

### CU-01: "Nombre del Caso de Uso"

[Se debe realizar una descripción de los escenarios del caso de uso.]

Iniciador	Actor que inicia el caso de uso.	
Otros actores	Otros actores, si existen.	
Precondiciones	Condiciones que deben cumplirse para que pueda realizarse el caso de uso.	
Flujo básico		
Actor	Sistema	
1. ...		
	2. ...	
	3. ...	
4. ...		
Flujo alternativo 1	Indicar otro camino por el cual se puede realizar el caso de uso, aparte del flujo básico	
Flujo alternativo 2		
...		

Flujo alternativo n	
Poscondiciones	Indicar los cambios que se producirán o los datos que deben quedar registrados.

[Cuando se ejecuta una instancia de un caso de uso, éste interactúa con instancias de actores y realiza una secuencia de acciones de acuerdo a lo que se describió en la especificación del caso de uso. Para mostrar la realización de un caso de uso se puede emplear: un *diagrama de clases* con todas las clases que participan en él; *diagramas de interacción* (secuencia y/o comunicación); o, como indica UWE, *diagramas de actividad*.

OBSERVACIÓN: UWE sugiere lo siguiente "Siguiendo el principio de utilizar UML siempre que sea posible para la especificación, se refinan los requerimientos con diagramas de actividad UML. Para cada caso de uso no trivial, se construye al menos un diagrama de actividad para el flujo principal de tareas a realizar para proporcionar la funcionalidad indicada por el caso de uso correspondiente. Opcionalmente, pueden utilizarse diagramas adicionales para excepciones y variantes."]

## CU-02: "Nombre del Caso de Uso"

...

...

## CU-nn: "Nombre del Caso de Uso"

## 4 DESCRIPCIÓN DEL DISEÑO

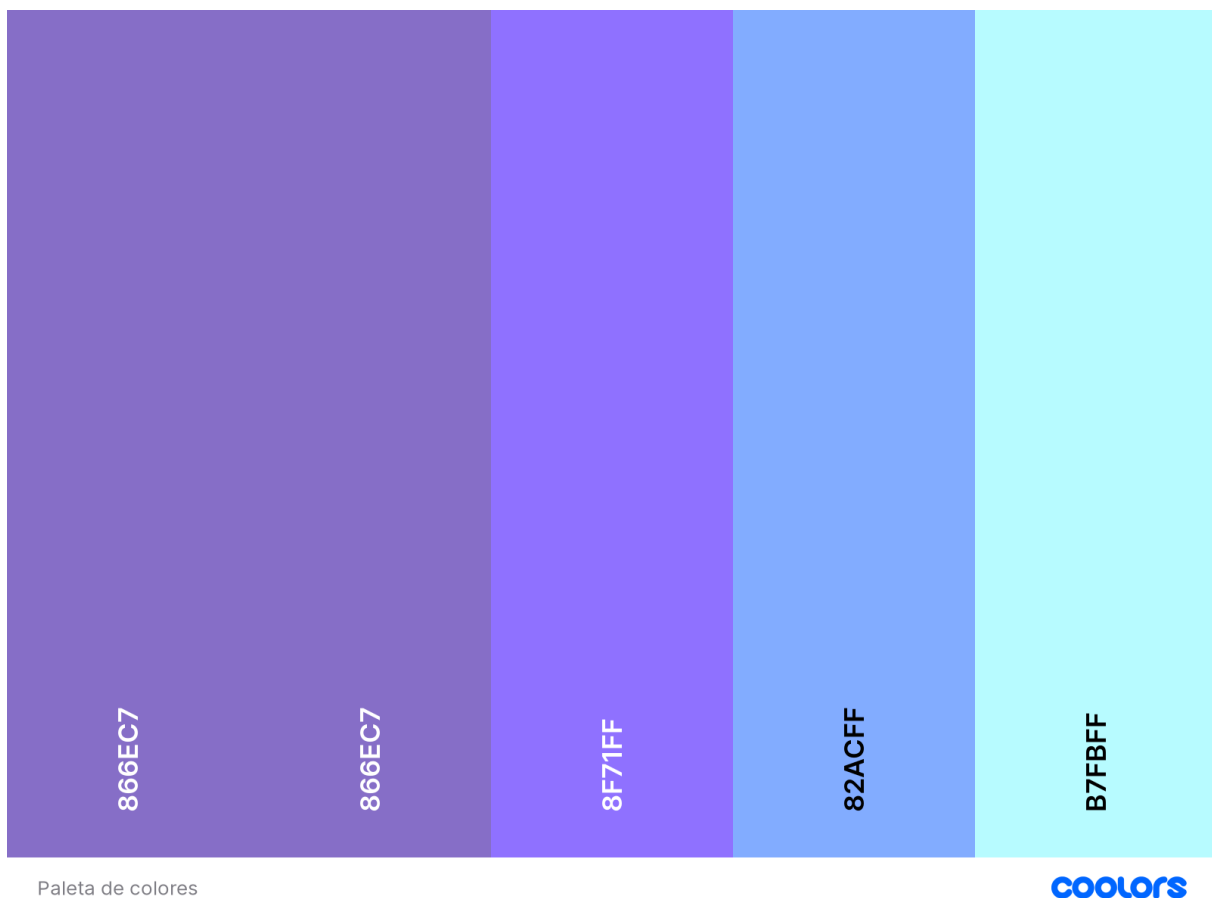
[Para el desarrollo de toda esta sección 4 utilizar como bibliografía de soporte:

(1) Schwinger, W.; Koch, N. "Modeling Web Applications", Chapter 3 en: Kappel, G.; Pröll, B.; Reich, S.; Retschitzegger, W. (Editors) *Web Engineering. The Discipline of Systematic Development of Web Applications*, John Wiley & Sons Ltd., 2006.

(2) Koch, N.; Knapp, A.; Zhang, G.; Baumeister, H. "UML-Based Web Engineering. An Approach Based on Standards", Chapter 7 en: Rossi, G.; Pastor, O.; Schwabe, D.; Olsina, L. (Editors) *Web Engineering: Modelling and Implementing Web Applications*, Springer-Verlag London Limited, 2008.]

### 4.1 Interfaz gráfica (Mockups)

## 4.2 Paleta de colores usada.



## 5 GESTIÓN DE LA CONFIGURACIÓN

Debe especificar los lenguajes de programación y el stack tecnológico que debe ser utilizado en la construcción de la solución.  
especificar el uso y estructura de los repositorios de código  
acceso y uso de la plataforma tecnológica en la nube con la que cuenta cada equipo  
a configuración y uso de los ambientes de desarrollo y producción.

## 6 PRUEBAS

### 6.1 Descripción de pruebas unitarias

## **6.2 Descripción de pruebas de aceptación**

## **7 GLOSARIO**

## **8 ANEXO(S)**

[ES OPCIONAL]

[Se pueden agregar anexos, si se consideran necesarios para obtener mayor claridad en el contenido del documento.]