|  |
| --- |
| DuocUC |
| Especificación de requisitos de software |
| *Proyecto: Gardenia Mobile* |
|  |
| **Revisión*: 1.2.1*** |
| **10/11/2024** |

|  |
| --- |
| Especificación de Requisitos según estándar de IEEE 830 para un Sistema Biblioteca. |

Contenido

[Ficha del documento 3](#_Toc440200284)

[1. Introducción 4](#_Toc440200285)

[1.1. Propósito 4](#_Toc440200286)

[1.2. Ámbito del Sistema 4](#_Toc440200287)

[1.3. Definiciones, Acrónimos y Abreviaturas 4](#_Toc440200288)

[1.4. Referencias 4](#_Toc440200289)

[1.5. Visión General del Documento 4](#_Toc440200290)

[2. Descripción General 5](#_Toc440200291)

[2.1. Perspectiva del Producto 5](#_Toc440200292)

[2.2. Funciones del Producto 5](#_Toc440200293)

[2.3. Características de los Usuarios 5](#_Toc440200294)

[2.4. Restricciones 5](#_Toc440200295)

[2.5. Suposiciones y Dependencias 5](#_Toc440200296)

[2.6. Requisitos Futuros 5](#_Toc440200297)

[3. Requisitos Específicos 6](#_Toc440200298)

[3.1. Interfaces Externas 6](#_Toc440200299)

[3.2. Funciones 6](#_Toc440200300)

[3.3. Requisitos de Rendimiento 6](#_Toc440200301)

[3.4. Restricciones de Diseño 6](#_Toc440200302)

[3.5. Atributos del Sistema 6](#_Toc440200303)

[3.6. Otros Requisitos 6](#_Toc440200304)

# Ficha del documento

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Revisión** | **Autor** | **Modificación** |
| *10-11-2024* | *1.1.1* | *Giovanna Espinosa* | *información Inicial* |
| *05-11-2024* | *1.2.1* | *Giovanna Espinosa* | *Modificación a requisitos funcionales* |

|  |  |  |
| --- | --- | --- |
| Por el cliente |  | Por la empresa suministradora |
| [Firma] |  | [Firma] |
| Sr./Sra. |  | Sr./Sra. |

# 1. Introducción

## 1.1. Propósito

El propósito de este documento es describir los requerimientos del sistema para el proyecto Gardenia, una plataforma móvil de comercio electrónico dedicada a la venta de productos de jardinería. Este documento proporciona una visión detallada de las funcionalidades, características, y restricciones del sistema, sirviendo como base para el desarrollo, pruebas, y mantenimiento del software.

## 1.2. Ámbito del Sistema

Gardenia es un sistema de comercio electrónico diseñado para permitir a los usuarios comprar productos de jardinería en línea. El sistema incluye funcionalidades para la gestión de productos, carrito de compras, gestión de direcciones de envío. Los usuarios pueden registrarse, iniciar sesión, gestionar sus compras y gestionar sus perfiles.

## 1.3. Definiciones, Acrónimos y Abreviaturas

* **ERS**: Especificación de Requerimientos de Software
* **API**: Interfaz de Programación de Aplicaciones
* **UI**: Interfaz de Usuario
* **CRUD**: Crear, Leer, Actualizar, Eliminar
* **CSRF**: Cross-Site Request Forgery

## 1.4. Referencias

* Documentación de Ionic: [https://ionicframework.com/docs/](https://docs.djangoproject.com/)
* Documentación de Bootstrap: <https://getbootstrap.com/>
* Documentación de Angulas: <https://docs.angular.lat/docs>
* Documentación Firebase: <https://firebase.google.com/docs>

## 1.5. Visión General del Documento

Este documento se organiza en tres secciones principales:

1. Introducción: Proporciona una visión general del propósito, ámbito, definiciones, referencias y estructura del documento.
2. Descripción General: Describe la perspectiva del producto, sus funciones, características de los usuarios, restricciones, suposiciones y dependencias, y requisitos futuros.
3. Requisitos Específicos: Detalla los requisitos de interfaces, funcionales y no funcionales, así como otros requisitos adicionales.

# 2. Descripción General

## 2.1. Perspectiva del Producto

## 2.2. Funciones del Producto

* **Gestión de Usuarios**: Registro, inicio de sesión, perfil de usuario, y suscripciones.
* **Gestión de Productos**: Visualización de productos, búsqueda y filtrado.
* **Carrito de Compras**: Adición y eliminación de productos, cálculo de totales y descuentos.
* **Procesamiento de Pagos**: Selección de direcciones, aplicación de cupones, confirmación y procesamiento de pagos.
* **Gestión de Pedidos**: Creación y almacenamiento de pedidos, actualización de stock.

## 2.3. Características de los Usuarios

* **Usuarios No Registrados**: Pueden navegar y buscar productos, pero necesitan registrarse para realizar compras.
* **Usuarios Registrados**: Pueden gestionar su perfil, agregar productos al carrito, seleccionar direcciones de envío, ver información de compras.

## 2.4. Restricciones

* **Tecnológicas**: El sistema debe desarrollarse utilizando Ionic, Angular, FireBase y Bootstrap.
* **Legales**: Cumplimiento con las leyes de protección de datos personales y comercio electrónico.
* **De Rendimiento**: El sistema debe ser capaz de manejar hasta 1000 usuarios concurrentes sin degradación significativa del rendimiento.

## 2.5. Suposiciones y Dependencias

* Se asume que los usuarios tienen acceso a internet y un navegador web moderno.
* El sistema depende de servicios externos para el procesamiento de pagos.
* Se asume que la base de datos estará disponible y será accesible en todo momento.

## 2.6. Requisitos Futuros

* **Mejoras de UI/UX:** Diseño adaptativo para diferentes dispositivos móviles y mejoras en la experiencia de usuario.
* **Expansión de Funcionalidades:** Integración con métodos de pagos, sistemas de personalización de búsqueda, filtros y orden de productos. Validación y autentificación de usuarios, recuperación de claves mediante correo electrónico. Guardado de información de métodos de pagos (asociar a tarjetas de debito/crédito)

# 3. Requisitos Específicos

## 3.1 Requisitos comunes de los interfaces

### 3.1.1 Interfaces de usuario

* **Registro e Inicio de Sección:** Formularios de registro y autentificación de usuarios al sistema.
* **Carrito de Compras:** Vista de carrito con opciones para actualizar cantidades, aplicar cupones y descuentos y proceder al pago.
* **Gestión de Perfil:** Formularios para actualizar información del usuario, gestión de direcciones de envio, gestión de suscripciones.
* **Catálogo de Productos:** Paginas para navegar productos.

### 3.1.2 Interfaces de hardware

* **Servidor:** Servidor web para alojar el sistema.
* **Cliente:** Dispositivos con navegación web (Pc, Tablet, Smartphone)

### 3.1.3 Interfaces de software

* **API de Pago:** Integración con servicios de pago externos.
* **API Maps:** Integración de API Maps para la gestión de direcciones.
* **Base de Datos:** Conexión a una base de datos relacional.

### 3.1.4 Interfaces de comunicación

* **HTTP/HTTPS:** Comunicación entre el cliente y el servidor.
* **Email:** Notificaciones y confirmaciones de pedidos.

## 3.2 Requisitos funcionales

### 3.2.1 Registro de Usuario

* **RF1.1:** El sistema debe permitir a los usuarios registrarse proporcionando nombre, email y contraseña
* **RF1.2:** El sistema debe enviar un correo electrónico de confirmación al usuario después de registrarse.

### 3.2.2 Inicio de Sesión

* **RF2.1:** El sistema debe permitir a los usuarios iniciar sesión con su email y contraseña.
* **RF2.2:** El sistema debe permitir a los usuarios restablecer su contraseña mediante un enlace enviado a su email registrado.

### 3.2.3 Gestión de Perfil

* **RF3.1:** El sistema debe permitir a los usuarios actualizar su contraseña.
* **RF3.2:** El sistema debe permitir a los usuarios gestionar sus direcciones, pudiendo ver, añadir, modificar o eliminar las direcciones.
* **RF3.3:**  El sistema debe permitir a los usuarios poder ver el historial de pedidos y el estado de estos desde su perfil.

### 3.2.4 Navegación y Búsqueda de Productos

* **RF4.1:**  El sistema debe permitir a los usuarios buscar productos utilizando productos claves.
* **RF4.2:** El sistema debe permitir a los usuarios filtrar los productos por categorías, subcategorías, precio y disponibilidad.
* **RF4.3:** El sistema debe mostrar los productos disponibles incluyendo, imágenes, descripción., precio y stock disponible.

### 3.2.5 Carrito de Compras

* **RF5.1:**  El sistema debe permitir a los usuarios agregar productos al carrito de compras.
* **RF5.2:**  El sistema debe permitir a los usuarios ver y actualizar las cantidades de los productos en el carrito de compras
* **RF5.3:**  El sistema debe permitir a los usuarios eliminar productos del carrito de compras
* **RF5.4:**  El sistema debe mostrar el total del carrito incluyendo los descuentos aplicables.

### 3.2.6 Procesamiento de Pagos

* **RF6.1:**  El sistema debe permitir a los usuarios seleccionar una dirección de envío antes de proceder con el pago.
* **RF6.2:**  El sistema debe procesar pagos a través de una integración con un servicio de pago externo.
* **RF6.3:**  El sistema debe enviar un correo electrónico de confirmación al usuario después de completar la compra

### 3.2.7 Gestión de Pedidos

* **RF7.1:**  El sistema debe permitir a los administradores ver una lista de todos los pedidos realizados.
* **RF7.2:**  El sistema debe permitir a los administradores actualizar el estado de los pedidos (pagado, preparando, enviado, en camino, entregado)
* **RF7.3:**  El sistema debe permitir a los usuarios ver el estado de sus pedidos en su perfil.

### 3.2.8 Gestión de Productos

* **RF8.1:**  El sistema debe permitir a los administradores agregar, actualizar y eliminar productos
* **RF8.2:**  El sistema debe permitir a los administrados gestionar el stock de los productos.

### 3.2.10 Notificaciones

* **RF10.1:**  El sistema debe permitir enviar notificaciones por correo electrónico a los usuarios en eventos clave, como registro, confirmación de pedido, y actualizaciones de estado del pedido.

**3.3 Requisitos no funcionales**

### 3.3.1 Requisitos de rendimiento

* **Tiempo de Respuesta:**  El sistema debe responder a las solicitudes de los usuarios en menos de 2 segundos en el 90% de las veces.

### 3.3.2 Seguridad

* **Autentificación y Autorización:** El sistema debe garantizar que solo los usuarios autentificados puedan acceder a funcionalidades protegidas.
* **Protección de Datos:** Los datos personales y de pago deben ser almacenados y transmitidos de manera segura.

### 3.3.3 Fiabilidad

* **Disponibilidad:**  El sistema debe estar disponible al menos el 97% del tiempo

### 3.3.4 Disponibilidad

* **Respaldo:**  El sistema debe contar con mecanismos de respaldo y recuperación ante problemas que afectan su funcionalidad.

### 3.3.5 Mantenibilidad

* **Documentación:** El código del sistema debe estar bien documentado para facilitar su mantenimiento.

### 3.3.6 Portabilidad

* **Compatibilidad de Navegadores:** El sistema debe ser compatible con los principales navegadores web y tener soporte mobile.

## 3.4 Otros Requisitos

* **Accesibilidad:** El sistema debe cumplir con la normas de accesibilidad web para asegurar que sea usable por personas con discapacidad.