| DUOC UC - ESCUELA DE INFORMÁTICA Y TELECOMUNICACIONES |
| --- |
| Especificación de Requisitos de Software |
| *Proyecto: Amasanderia Masamagna* |
|  |
| **Revisión*: [99.99]*** |
| **[28/10/2025]** |

| Especificación de Requisitos según estándar de IEEE 830. |
| --- |

Integrantes:

-Camila Pino

-Gonzalo Navarrete

**Contenido**

[**1. Introducción 3**](#_pgr5v1r2fxr)

1.0[. 3](#_o6fyt2mwrsk6)

[1.1. Propósito 3](#)

[1.2. Ámbito del Sistema 3](#_7cqip9gmp3xl)

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

[1.4. Referencias 4](#_vk0yxuem4pxg)

[1.5. Visión General del Documento 4](#_2pvx4yr8a1sl)

[**2. Descripción General 5**](#_3dlrdmptsrz7)

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

[2.2. Funciones del Producto 6](#_tdrv9uo4szdt)

[2.3. Características de los Usuarios 7](#_yhxm9xvvb2qd)

[2.4. Restricciones 7](#_5aosut7h9rxt)

[2.5. Suposiciones y Dependencias 7](#_1nzzpx8obejz)

[2.6. Requisitos Futuros 7](#_tewdn6em7dk4)

[**3. Requisitos Específicos 8**](#_dl7y1n28idpp)

[3.1 Requisitos comunes de las interfaces 8](#_6li9jmf4i1l5)

[*3.1.1 Interfaces de usuario 8*](#_qmb1abp4ps42)

[*3.1.2 Interfaces de hardware 8*](#_f0oizpnz4jvo)

[*3.1.3 Interfaces de software 8*](#_yjpydmtbs0ot)

[*3.1.4 Interfaces de comunicación 8*](#_bgc8fqka0usu)

[3.2 Requisitos funcionales 9](#_bgroarlctquw)

[3.3 Requisitos no funcionales 9](#_vp5hwlgelpxy)

[*3.3.1 Requisitos de rendimiento 9*](#_3kdh413o4hlo)

[*3.3.2 Seguridad 9*](#_7hzmow7ra6zb)

[*3.3.3 Fiabilidad 9*](#_poky4yx07uha)

[*3.3.4 Disponibilidad 9*](#_ggaievln7uwc)

[*3.3.5 Mantenibilidad 10*](#_qcce6finbx9t)

[*3.3.6 Portabilidad 10*](#_wi6h9uelnqmr)

[3.4 Otros Requisitos 10](#_ukp50sxafeen)

# 1. Introducción

## Este documento constituye la Especificación de Requisitos Software (ERS) del sitio web “Amasanderia Masamagna”. En este documento, también se detalla el propósito del proyecto, el ámbito del sistema, definiciones y acrónimos, las referencias utilizadas y los requisitos funcionales, no funcionales y posibles futuras integraciones.

## 1.1. Propósito

El propósito de este documento es especificar los requisitos de software para el desarrollo del sitio web de la pastelería “Amasandería Masamagna”, además de estar dirigido a los propietarios de la amasandería como medio de validación de los requisitos y los desarrolladores sirviendo como guía del proyecto.

## 1.2. Ámbito del Sistema

El sistema implementado es una aplicación web full-stack completa:

**Tecnologías Implementadas:**

* Frontend: React con componentes para Productos, Contacto, Login, Carrito.
* Backend: Spring Boot con API REST (Producto, Usuario, Carrito, ItemCarrito).
* Base de Datos: MySQL gestionada con XAMPP y MySQL Workbench.
* Comunicación: REST API entre React y Spring Boot.

**Funcionalidades implementadas:**

* Catálogo de productos con imágenes, descripciones y precios reales.
* Sistema de autenticación y registro funcional.
* Carrito de compras persistente en base de datos.
* CRUD completo de Producto y Usuario (Crear, Listar, Actualizar, Eliminar, Desactivar)
* Gestión de usuarios con roles diferenciados.
* API REST documentada con Swagger.
* Interfaz React moderna y responsive.

**Lo que hace por ahora:**

* Persiste todos los datos en base de datos MySQL.
* Gestiona pedidos reales con ítems de carrito.
* Autentica usuarios contra base de datos
* Administra inventario de productos en tiempo real.

**Lo que no hace por ahora:**

* No procesa pagos en línea.

**Los beneficios del sistema son:**

* Ampliar el alcance comercial de la pastelería.
* Facilitar la visibilidad de productos.

## 1.3. Definiciones, Acrónimos y Abreviaturas

ERS: Especificación de Requisitos de Software.

Frontend: Interfaz visual y navegable por el usuario.

Backend: Lógica de negocio y base de datos que se integrará en futuras versiones.

Carrito: Sección del sistema donde se agrupan productos seleccionados por el cliente antes de confirmar la compra.

Login: iniciar sesión en el sitio web.

Register: Registrarse en el sitio web.

SPA: Single Page Application (Aplicación de una sola página).

API REST: Interfaz de programación de aplicaciones RESTful.

JPA: Java Persistence API (Persistencia de datos en Java).

CRUD: Create, Read, Update, Delete (Operaciones básicas de datos).

XAMPP: Entorno de desarrollo web con Apache, MySQL, TomCat.

## 1.4. Referencias

En esta subsección se mostrará una lista completa de todos los documentos referenciados en la ERS.

## 1.5. Visión General del Documento

El documento se organiza de la siguiente manera:

1.- Introducción.

2.- Descripción general del producto y los usuarios.

3.- Especificación de los requisitos (Funcionales y no Funcionales)

# 

# 2. Descripción General

El desarrollo del sistema depende de factores técnicos, de negocio, de usuario y de entorno. Técnicamente, se construirá un sitio web frontend con HTML, CSS y JavaScript, compatible con distintos dispositivos y navegadores. Los usuarios principales serán clientes finales, que explorarán productos y simularán pedidos, vendedor, que simulara gestionar pedidos, agregar productos y usuarios y un administrador que tendrá acceso total al sistema .

El objetivo del sistema es aumentar la visibilidad de la pastelería, mejorar la experiencia del cliente y preparar el camino hacia un sitio web completo. En esta primera etapa, se limita a exhibir productos y simular compras, sin base de datos ni procesamiento real de pedidos.

## 2.1. Perspectiva del Producto

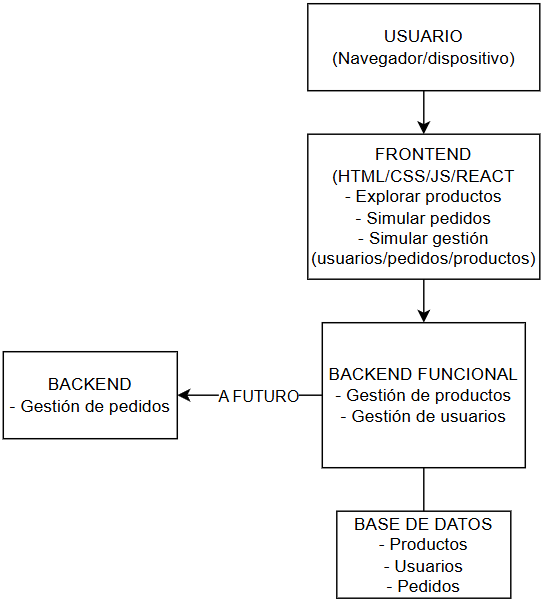
El sistema implementado es una aplicación web completa que ya incluye backend y base de datos.

**Arquitectura implementada:**

* Frontend React: Aplicación moderna con componentes para Productos, Contacto, Login y Carrito.
* Backend Spring Boot: API REST completa con controladores, servicios, repositorios y entidades (Producto, Usuario, Carrito, ItemCarrito)
* Base de datos MySQL: Persistencia real mediante XAMPP y MySQL Workbench.
* Comunicación: Frontend (puerto 5173) consume API del backend mediante HTTP/REST.

**Funcionalidades:**

* Autenticación real con usuarios en base de datos.
* Carrito persistente que mantiene los ítems entre sesiones.
* CRUD completo de productos con operaciones en base de datos.
* Gestión administrativa real de usuarios y pedidos.
* API documentada con Swagger/OpenAPI.



## 2.2. Funciones del Producto

**Resumen de funciones:**

- Visualización de catálogo de productos.

- Visualización de precios y sabores.

- Sección de contacto/información de la pastelería

- Catálogo interactivo de productos con imágenes y precios

- Carrito de compras persistente en base de datos

- Sistema de autenticación y roles (Cliente, Administrador)

- Panel admin con gestión completa (productos, usuarios, pedidos)

- APIs REST para todas las operaciones del sistema

## 2.3. Características de los Usuarios

* Cliente: Navega, agrega productos al carrito, realiza pedidos
* Administrador: Gestiona productos, usuarios, pedidos desde panel admin
* Vendedor: Visualiza y gestiona órdenes de pedidos

## 2.4. Restricciones

Las restricciones del sistema son:

* Desarrollado en HTML, CSS, JavaScript y React.
* Backend: Spring Boot, Java 11.
* Base de datos: MySQL con XAMPP.
* Entorno: LocalHost (React: 5173. Spring Boot: 8080)
* Persistencia: Base de datos real MySQL.
* Lenguaje: Español.

## 2.5. Suposiciones y Dependencias

Las suposiciones y dependencias son:

* Se supone que los usuarios tendrán acceso a internet y a navegadores web.
* Java y [Node.js](http://node.js) instalados para desarrollo.
* XAMPP ejecutándose para base de datos MySQL.
* Conexión a internet para consumo de APIs.

## 2.6. Requisitos Futuros

Dentro de los requisitos futuros, tenemos:

* Integración con pasarelas de pago (Webpay, MercadoPago).
* Sistema de notificaciones por email.
* Panel de reportes y analytics avanzados.
* Aplicación móvil complementaria.

# 3. Requisitos Específicos

En esta sección se describe los requisitos técnicos detallados para el desarrollo de la aplicación web full-stack, incluyendo especificaciones para el frontend React, backend Spring Boot y base de datos MySQL, asegurando una implementación coherente entre todos los componentes del sistema.

## 3.1 Requisitos comunes de las interfaces

Descripción detallada de todas las entradas y salidas del sistema de software, incluyendo la comunicación entre el frontend React, backend Spring Boot y base de datos MySQL.

### 3.1.1 Interfaces de usuario

* Tecnología: Aplicación React con componentes.
* Header/Navegación Superior: Menú con acceso a Productos, Contacto, Login y Carrito.
* Panel de Administración: Sidebar con opciones (Panel Principal, Productos, Usuarios, Pedidos, Configuración, Cerrar Sesión)
* Diseño: Interfaz responsive con temática de pastelería.
* Navegación: React Router para transiciones suaves entre secciones sin recarga de página.
* Área de Contenido Dinámico: Sección central que cambia según la ruta:
* Catálogo de Productos: Grid de tarjetas con imágenes, nombres, precios.
* Detalle de Producto: Vista individual con descripción completa.
* Carrito de Compras: Lista de productos seleccionados con cantidades y total.
* Formulario de Contacto: Campos para nombre, email y mensaje.
* Panel Administrativo: Vistas de gestión para productos, usuarios y pedidos.

### 3.1.2 Interfaces de hardware

* Spring Boot - MySQL: Conexión JDBC con entidades JPA
* React - Spring Boot: Comunicación HTTP/REST con JSON
* Swagger/OpenAPI: Documentación automática de endpoints
* Navegador - React: Ejecución de SPA con componentes

### 3.1.3 Interfaces de software

Indicar si hay que integrar el producto con otros productos de software.

* Spring Boot - MySQL: Conexión JDBC con entidades JPA.
* React - Spring Boot: Comunicación HTTP/REST con JSON.
* Swagger/OpenAPI: Documentación automática de endpoints.
* Navegador - React: Ejecución de SPA con componentes.

### 3.1.4 Interfaces de comunicación

* Protocolo: HTTP/HTTPS para APIs REST.
* Formato: JSON para intercambio de datos.
* Origen: Configurado entre React (5173) y Spring Boot (8080).
* Autenticación: Headers HTTP para gestión de sesiones.

## 3.2 Requisitos funcionales

**Cliente:**

* Requisito Funcional-01: Visualizar catálogo de productos con imágenes y precios
* Requisito Funcional-02: Ver detalle completo de productos (descripción, ingredientes)
* Requisito Funcional-03: Gestionar carrito de compras (agregar, modificar, eliminar)
* Requisito Funcional-04: Proceso de checkout y generación de pedidos
* Requisito Funcional-05: Formulario de contacto funcional

**Administrador:**

* Requisito Funcional-06: Panel de administración con navegación completa
* Requisito Funcional-07: CRUD de productos (crear, listar, editar, eliminar, desactivar)
* Requisito Funcional-08: Gestión de usuarios del sistema
* Requisito Funcional-09: Administración de pedidos y cambios de estado
* Requisito Funcional-10: Configuración general del sistema
* Requisito Funcional-11: Control de sesiones (login/logout seguro)

**Sistema:**

* Requisito Funcional-12: Autenticación y autorización por roles
* Requisito Funcional-13: Comunicación frontend-backend mediante APIs REST
* Requisito Funcional-14: Persistencia de datos en base de datos MySQL

## 3.3 Requisitos no funcionales

* Requisito No Funcional-01: Carga de página principal en < 3 segundos (conexión 10Mbps)
* Requisito No Funcional-02: Compatibilidad con navegadores modernos
* Requisito No Funcional-03: Código modular y mantenible (React components + Spring services)
* Requisito No Funcional-04: APIs RESTful documentadas con Swagger
* Requisito No Funcional-05: Interfaz responsive en dispositivos móviles y desktop
* Requisito No Funcional-06: Persistencia confiable en base de datos MySQL

### 3.3.1 Requisitos de rendimiento

- El sitio debe cargar en menos de 3 segundos en una conexión promedio de 10 Mbps.

### 3.3.2 Seguridad

- No se manejan datos sensibles en esta segunda versión.

### 3.3.3 Fiabilidad

- El sistema debe estar disponible un 95% del tiempo.

### 3.3.4 Disponibilidad

- Accesible desde cualquier navegador moderno.

### 3.3.5 Mantenibilidad

* Código estructurado en módulos (HTML/CSS/JS)

### 3.3.6 Portabilidad

* Compatible con navegadores: Chrome, Firefox, Edge, Safari.

## 3.4 Otros Requisitos

* Diseño visual coherente con identidad de pastelería
* Experiencia de usuario intuitiva (máximo 3 clics para acciones principales)
* Documentación técnica completa para desarrolladores