



COMPUTER SCIENCE DEPARTMENT

SOFTWARE ENGINEERING

Advanced web programming - NRC 27819

Project Backlog

Group N°5

Lainez Suárez Ricardo Andrés

Llumiyinga Ñacato Ariel

Santi Cruz Jeancarlo Javier

Professor: Edison Lascano

date: 20/10/2025

Software requirements specification:

1. Introduction

1.1. Purpose

The purpose of this document is to define the functional and non-functional requirements for the **FastFood Web Application**, a responsive and user-friendly system designed to help a fast food restaurant increase its sales and expand its customer reach.

The web application will allow customers to explore the restaurant's full catalog of meals and beverages, customize their orders according to preferences or allergies, and place online orders for delivery or pickup.

The web application consists of customer service, offering quality products, orderly product lists, daily specials, and the ability to make reservations.

1.2. Scope

The **FastFood Web Application** aims to modernize the restaurant's business model by providing an online platform that enhances customer interaction and improves the overall purchasing experience.

The system will enable users to:

- Browse the food and drinks catalog with images and prices.
- Customize meals by adding or removing ingredients (especially for allergy concerns).
- View payment methods and process secure online transactions.
- Read and post customer reviews and ratings.

1.3. Definitions, Acronyms, and Abbreviations

- **CRUD:** Create, Read, Update, Delete
- **UI:** User Interface
- **UX:** User Experience
- **API:** Application Programming Interface
- **DBMS:** Database Management System

2. Overall Description

2.1. Product Purpose and Context

Main Objective of the System

The goal is to increase restaurant sales and profitability by establishing an online platform that improves customer interaction. This will enable the restaurant to compete with market trends.

Role of the Application

La aplicación es un sistema web responsive y fácil de usar diseñado para ser el canal principal de servicio al cliente. Proporcionará productos de calidad, listas de productos ordenadas, especiales diarios y la capacidad de hacer reservaciones

2.2. Scope and Boundaries

High-Level Features

The system will allow customers to browse the complete food and beverage catalog, customize orders according to preferences or allergies, and process secure online transactions. In addition, it will integrate a digital sales record and an administration panel.

Platform Requirements

The solution is a web application, which means that development must ensure compatibility with modern browsers and be responsive to provide a consistent user experience on mobile and desktop devices.

2.3. User characteristics

Profile	Role in the System	Key Expectations
Customer / Guest	Primary user who navigates the menu, customizes meals, places orders, and processes payment. They can also leave reviews.	Look for convenience, ease of use, and a fast, visually appealing shopping experience on any device.

FAST FOOD WEB APPLICATION - FEATURES PLAN

Feature	Objective	Key User Stories	Acceptance Criteria	Main APIs / Data
1. Digital Menu Catalog	Display the complete food and drinks catalog with images, prices, and filters.	<ul style="list-style-type: none">• As a customer, I want to view the full menu with images and prices to decide what to order.• As a customer, I want to filter by category or tags such as vegan or spicy.	<ul style="list-style-type: none">• When visiting '/menu', the page shows product cards with name, price, and photo.• Filters correctly display selected categories.	GET /api/products GET /api/products?category= DB: products(id, name, description, price, image_url, category, tags[])

2. Meal Customization & Allergies	Allow users to add/remove ingredients and store allergy preferences.	<ul style="list-style-type: none"> • As a customer, I want to remove ingredients if I have allergies. • As a customer, I want to add extras (cheese, sauce) to personalize meals. 	<ul style="list-style-type: none"> • A modal lets users mark ingredients to remove or add. • Prices recalculate dynamically. • Allergens generate system warnings. 	GET /api/products/:id/config POST /api/users/:id/allergies DB: ingredients, product_ingredients, user_allergies
3. Cart & Checkout	Provide end-to-end order management (pickup or delivery).	<ul style="list-style-type: none"> • As a customer, I want to add products to my cart and checkout. • As a staff member, I want to see active orders and update their status. 	<ul style="list-style-type: none"> • Cart shows totals and order summary. • Order stored in database with status 'Pending'. 	POST /api/cart POST /api/orders GET /api/orders/:id DB: carts, orders, order_items
4. Payments, Promotions & Loyalty	Enable secure online payments and rewards for frequent customers.	<ul style="list-style-type: none"> • As a customer, I want to pay online securely. • As a customer, I want to use discount coupons or loyalty points. 	<ul style="list-style-type: none"> • Coupon codes apply discounts before payment. • Successful payments change order to 'Approved' state. 	POST /api/payments POST /api/promos/validate GET /api/loyalty DB: payments, promos, loyalty

5. Admin Dashboard	Allow administrators to manage inventory, reviews, and view analytics.	<ul style="list-style-type: none"> • As an admin, I want to see daily sales, best-selling items, and stock levels. • As an admin, I want to moderate customer reviews. 	<ul style="list-style-type: none"> • Dashboard displays KPIs and sales graphs. • Low-stock alerts appear when below threshold. 	GET /api/admin/metrics PATCH /api/inventory GET /api/reviews DB: inventory, reviews, reports
--------------------	--	--	--	---