Project proposal

This project represents a digital catalogue of goods for a supermarket, designed to provide structured, up-to-date information about products available in the store.

The system aims to simplify product discovery for customers and product management for store staff by centralizing all product-related data in a single application

The catalogue allows users to browse products by categories, search by name or barcode, and apply filters such as price range, brand, and availability

From the staff perspective, the system provides tools to manage products, prices, and stock quantities efficiently

The project is developed as a monolithic application using the GO programming language, focusing on clarity of architecture, data modeling, and responsibility separation

Project relevance

The Real-Life Problem

In large supermarkets, customers often arrive with a specific goal. Despite the store having the item, shoppers face several hurdles:

Time Loss: Wasting time walking through endless aisles or searching multiple shelves.

Information Gap: Relying on busy employees for stock checks.

Inaccuracy: Products being out of stock, placed in the wrong category, or located in non-obvious aisles.

Manual Overhead: Staff must manually update prices and respond to repetitive inquiries.

The Go-Based Solution

This application acts as a "digital store map + live inventory helper." By leveraging Go’s efficiency in handling real-time data, the system addresses these issues by:

Instant Search: Customers scan a barcode or search by name to see exact availability

Precise Localization: The system displays the specific shelf/aisle location

Live Updates: Staff can update stock and locations instantly, ensuring the "source of truth" is always accurate

Competitor Analysis

In the current market, several platforms attempt to bridge the gap between supermarkets and digital users, yet they often fall short of providing a perfect local solution. Services like arbuz.kz or airba fresh, for instance, offer polished interfaces for online grocery shopping, but they operate primarily as warehouse-based delivery models rather than tools designed to reflect the live inventory of a specific physical storefront. Similarly, the magnum mobile app provides a loyalty system and product listings, but users often encounter discrepancies between the digital catalogue and the actual availability on the shelves due to delayed data synchronization

This project addresses these specific limitations by offering a dedicated, high-performance catalogue system. Unlike broad delivery-focused platforms, this application is built to be the "source of truth" for a single supermarket or chain. Developed with the go programming language for maximum efficiency, it ensures that stock levels, price changes, and product details are updated instantly. By removing the overhead of complex ogistics and delivery tracking, the system provides a lean, fast, and highly accurate tool that simplifies life for both the store staff and the everyday shopper

Target Users

1.Customers

The primary beneficiaries are local customers who value their time and seek a frictionless shopping experience. These users rely on the application to plan their visits by checking real-time product availability and comparing prices from the comfort of their homes. Whether it's a parent looking for a specific baby formula brand or a budget-conscious shopper filtering for the best deals, the system empowers them with a searchable, up-to-date digital aisle that eliminates the frustration of arriving at a store only to find an empty shelf

2.Store management and staff

From an operational perspective, the application serves as a vital tool for supermarket employees and floor managers. Unlike outdated paper-based systems or slow legacy software, this platform allows staff to efficiently update stock quantities, adjust prices instantly, and organize product categories. By centralizing all product-related data, the system reduces the manual workload for employees, minimizes human error in pricing, and ensures that the information provided to the customer always matches the reality of the warehouse

Planned features

To ensure a robust and functional MVP, the development will focus on the following core features:

Dynamic Product Search

A high-performance search engine that allows users to find products by name or category. This will demonstrate Go's efficiency in handling data filtering and string processing

Real-Time Price & Stock Updates

A secure management interface for staff where price changes and stock quantities can be updated. These changes will reflect instantly across the entire application without requiring a system restart

Category-Based Navigation

A structured browsing system that organizes goods into logical groups like Dairy, Bakery, Household, making it easier for users to explore the store's assortment

Availability Status

A simple visual indicator for each product ,for example "In Stock", "Low Stock", or "Out of Stock", to help customers plan their purchases based on current warehouse data

Orders History

A complete order history module that stores and displays all past customer orders.Orders are stored from the most recent to the oldest by date, and each entry includes the total price and a clear list of ordered items.This feature demonstrates structured data modeling in Go and efficient sorting or filtering logic

Real-Time Current Order Tracking

A real-time order tracking indicator displayed as a dedicated icon. The icon shows the status of the user`s current active order and updates instantly

without page reloads.This highlights real-time state updates across the application and improves user experience by making order progress visible at all times

Modules

The system is divided into several logical modules, each handling a specific part of the functionality:

- Product Module – handles everything related to products: searching, filtering, viewing details, and categories. It ensures that customers can quickly find the products they need and see up-to-date availability
- Order Module – manages orders: keeps order history, shows the current order and its status. This module allows customers to always know what they have ordered and when
- User Module – manages user accounts for both customers and staff. It handles authentication and basic user information.
- Staff Management Module – provides staff tools for managing the store: adding new products, updating prices and stock quantities, and organizing categories. This module keeps the catalog accurate and reduces manual work

Data Flow

The system is designed so that all product and order information is always up-to-date and available in real time

Example 1: Customer searching for a product

1. The customer enters a query in the interface
2. The search module receives the query and filters products by name, category, price, brand, and availability

3. The data layer retrieves matching records from the database and returns them
4. The interface displays the product list with detailed information and availability status

Example 2: Staff updating product information

1. Staff updates a product's price or stock quantity through the management panel
2. The product management module validates the data and updates the database
3. Changes are instantly reflected in the customer interface, ensuring accurate prices and stock levels

Responsibilities

Clear separation of responsibilities ensures efficiency and usability for both customers and staff

Customers:

- Browse and filter products by category, brand, price, and availability
- View product details and current availability status
- Track the current order and access order history

Staff:

- Add new products and update existing ones
- Manage product prices and stock levels
- Organize products into categories and ensure the catalog is accurate

System (Application Layer):

- Ensures that product and order data is always accurate and up-to-date
- Handles business logic for search, filtering, and product updates

- Maintains synchronization between the user interface and the database so that users always see real-time information