Software Requirements Specification

For

Smart Label System for Super Market Sector in TRNC

Version 1.0 approved

Prepared by

Yaman Küpeli(Team Leader)21000155

Kaan Sulkalar 20000063

Göktuğ Fırat Çelik 19000042

Mert Kaan Alpan 18000058

28/11/2024

# Table of Contents

[Table of Contents 1](#_Toc192725774)

[A.1 Preliminary Project Information 2](#_Toc1924967976)

[A.1.1 Project Details 2](#_Toc1834202818)

[A.1.2 Project Manager 2](#_Toc1865253643)

[A.2 Group Information 2](#_Toc2137267536)

[A.2.1 Team Members 3](#_Toc1832156832)

[Introduction 3](#_Toc1201572994)

[Purpose 4](#_Toc2099419648)

[Document Conventions 4](#_Toc1385258633)

[Intended Audience and Reading Suggestions 4](#_Toc535013333)

[Product Scope 4](#_Toc1184747840)

[Overall Description 5](#_Toc14503464)

[Product Perspective 5](#_Toc413681050)

[Product Functions 5](#_Toc1021487166)

[User Classes and Characteristics 6](#_Toc1129811769)

[Operating Environment 6](#_Toc1681791091)

[Design and Implementation Constraints 7](#_Toc1076330883)

[User Documentation 8](#_Toc1714019947)

[Assumptions and Dependencies 9](#_Toc485728839)

[Gantt Chart Analysis 10](#_Toc1990000017)

[External Interface Requirements 11](#_Toc1600221602)

[User Interfaces 11](#_Toc1039204989)

[Management Dashboard 11](#_Toc538597552)

[ Electronic Shelf Labels (ESLs) 11](#_Toc55783436)

[ Accessibility Features 12](#_Toc1451351333)

[Hardware Interfaces 12](#_Toc11323702)

[Electronic Shelf Labels (ESLs) 12](#_Toc280046751)

[2. Central Server Hardware 12](#_Toc853510346)

[Software Interfaces 13](#_Toc1138978253)

[Software Components 13](#_Toc1697904632)

[ Key Implementation Constraints 15](#_Toc796850920)

[Communications Interfaces 15](#_Toc514778277)

[System Features 17](#_Toc426139646)

[Real-Time Price Updates 17](#_Toc2086020069)

[Description and Priority 17](#_Toc2057791603)

[Stimulus/Response Sequences 17](#_Toc1392017735)

[Functional Requirements 17](#_Toc1524927402)

[Comprehensive Product Information 18](#_Toc1050123089)

[Description and Priority 18](#_Toc625837054)

[Stimulus/Response Sequences 18](#_Toc505720049)

[Functional Requirements 18](#_Toc1029698080)

[Threshold Alerts 18](#_Toc57233632)

[Description and Priority 18](#_Toc82553985)

[Stimulus/Response Sequences 18](#_Toc177873849)

[Functional Requirements 19](#_Toc2025383404)

[Customer Engagement Features 19](#_Toc1375052118)

[Description and Priority 19](#_Toc372777149)

[Stimulus/Response Sequences 19](#_Toc1727223869)

[Functional Requirements 19](#_Toc1669734776)

[Data Analytics and Reporting 20](#_Toc811687917)

[Description and Priority 20](#_Toc91070381)

[Stimulus/Response Sequences 20](#_Toc1202079866)

[Functional Requirements 20](#_Toc122734642)

[Energy Efficiency 20](#_Toc1599630064)

[Description and Priority 20](#_Toc1172882780)

[Stimulus/Response Sequences 20](#_Toc1481766165)

[Functional Requirements 21](#_Toc1194037)

[Integration with Store Systems 21](#_Toc1139468021)

[Description and Priority 21](#_Toc1393326043)

[Stimulus/Response Sequences 21](#_Toc99369092)

[Functional Requirements 21](#_Toc1195474336)

[Compliance with Regulatory Standards 21](#_Toc588974806)

[Other Nonfunctional Requirements 22](#_Toc217696853)

[Scalability 22](#_Toc983670544)

[Usability 22](#_Toc275746487)

[Availability 23](#_Toc1137730543)

[Compatibility 23](#_Toc945225112)

[Performance Requirements 23](#_Toc1575932549)

[Response Time 23](#_Toc1699885961)

[Data Throughput 23](#_Toc58198495)

[Energy Efficiency 23](#_Toc2104369937)

[Real-Time Analytics 24](#_Toc1656752908)

[Safety Requirements 24](#_Toc1349189118)

[Hardware Safety 24](#_Toc2086489130)

[Data Accuracy 24](#_Toc154032094)

[Regulatory Compliance 24](#_Toc147428010)

[Customer Safety 24](#_Toc1056350043)

[Security Requirements 24](#_Toc434158833)

[Data Protection 24](#_Toc2045636314)

[Authentication 25](#_Toc1992880325)

[Intrusion Detection 25](#_Toc2147018488)

[Network Security 25](#_Toc684497979)

[Software Quality Attributes 25](#_Toc499750004)

[Reliability 25](#_Toc1211895636)

[Maintainability 25](#_Toc1741768984)

[Interoperability 26](#_Toc342436051)

[Usability 26](#_Toc1955052224)

[Testability 26](#_Toc578945083)

[Portability 26](#_Toc959350914)

[Business Rules 26](#_Toc1357284671)

[Role-Based Access Control 26](#_Toc1887966468)

[Compliance with Retail Policies 27](#_Toc86334380)

[Customer Interaction 27](#_Toc199982547)

[Inventory Accuracy 27](#_Toc725117410)

[Analytics and Reporting 27](#_Toc1134637202)

[Other Requirements 27](#_Toc1280616348)

[Database Design 27](#_Toc716046460)

[Appendix A: Glossary 31](#_Toc489586490)

[APPENDIX B : Relevant Diagrams 34](#_Toc1979519031)

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Mert Kaan Alpan | 30/12/24 | Formatting | Rev.1B |
|  |  |  |  |

# A.1 Preliminary Project Information

## A.1.1 Project Details

|  |  |
| --- | --- |
| Project No | 6 |
| Project Name | Smart Label System for Supermarkets in TRNC |
| Start Date | October 10, 2024 |
| End Date | October 10, 2025 |
| Duration | 1 Year |

## A.1.2 Project Manager

|  |  |
| --- | --- |
| Name | Yaman Küpeli |
| ID No | 21000155 |
| Role | Project Manager / Software Developer |
| Address | Famagusta, TRNC |
| Email | [21000155@emu.edu.tr](mailto:21000155@emu.edu.tr) |

# 

## A.2 Group Information

### A.2.1 Team Members

|  |  |  |
| --- | --- | --- |
| Name | Role | Contact |
| Mert Kaan Alpan | Embedded Systems Developer / Hardware Designer | [18000058@emu.edu.tr](mailto:18000058@emu.edu.tr) |
| Kaan Sulkalar | UI/UX Designer / Frontend Developer | [20000063@emu.edu.tr](mailto:20000063@emu.edu.tr) |
| Goktüğ Firat Çelik | Data Management Specialist / Backend Developer | [19000042@emu.edu.tr](mailto:19000042@emu.edu.tr) |
| Yaman Kupeli | Project Manager / Software Developer | [21000155@emu.edu.tr](mailto:21000155@emu.edu.tr) |

# Introduction

## Purpose

The project estimates a complete electronic shelf labeling system that will allow an intelligent solution for real-time monitoring and updating of prices of merchandise around retail settings. The aim of the Smart Label System is to ensure efficiency in operation whereby for every fluctuation in product price, the retailer shall be informed in real time, at the same time warning the consumer in case such products exceed regulated pricing limits, hence accuracy in prices according to regulations or legislation for better consumers' shopping experiences.

## Document Conventions

This document is prepared as a standard IEEE Software Requirements Specification, applying MLA formatting styles. Section and subsection headings are highlighted in bold text and appropriately indented. All other content is presented in Times New Roman, 11-point font, with standard spacing for clear comprehension..

## Intended Audience and Reading Suggestions

This document targets all developers, project managers, and testers, as well as stakeholders of the Smart Label System project. The system and software requirements related to the project are presented in this document. It is suggested that the reader begin with the Introduction for an overview and then proceed further into sections relevant to the role for which a particular reader may be seeking details.

## Product Scope

Its Smart Label System is a software that enables real-time monitoring and updating of product prices in retail stores through electronic shelf labels. This system is expected to increase operational efficiency, provide price check alerts, and regulation compliance when prices exceed legal parameters. Such aims go hand-in-hand with companies wanting to increase customer satisfaction, cut down on labor-related costs and stay relevant by investing in new technologies.

# Overall Description

## Product Perspective

Derived from the Smart Label System, we present a stand-alone product that is focused on creating better price management in the retail environment. It deploys electronic shelf labels to dynamically monitor and refresh prices of products. It is also an independent solution which does not require strict integration with current inventory systems, so it would free the current methods of price control and regulation process and eliminate the inaccuracies and inconsistencies between systems. This system changes the pricing of the food items and also the labeling of food items from the level of ancient times to the next level of these days.

## Product Functions

The Smart Label System offers the following major functions:

 **Real-Time Price Updates:** Prices on electronic shelf labels are updated instantly, keeping everything across the store current and consistent.

 **Price Monitoring:** The system constantly checks product prices to make sure they match the central database—accuracy guaranteed.

 **Regulatory Compliance Alerts:** If a price goes over a regulated limit, the system sends alerts so retailers can stay within the law.

 **User Management Dashboard:** Retailers get a simple, intuitive dashboard to manage prices, promotions, and system settings all in one place.

 **Data Integration:** Works seamlessly with your existing inventory and point-of-sale systems for smooth, synchronized data management.

## User Classes and Characteristics

The Smart Label System is designed for these key users:

* **Store Managers:** They oversee everything—pricing, store operations, and more. They need full access to the system.
* **Pricing Staff:** These are the folks who handle daily price updates and promotions, keeping things fresh and accurate.
* **IT Administrators:** They take care of the technical side, like system maintenance and updates.
* **Sales Associates:** They only need basic access to quickly check and verify product prices.
* **Compliance Officers:** Their job? Make sure all pricing stays within legal limits.

The **primary users** are Store Managers and Pricing Staff since they’re the ones interacting with the system the most.

.

## Operating Environment

Smart Label System is designed to integrate seamlessly in retail store environments In order to operate effectively, it requires the following:

**Hardware Platform**

* Electronic Shelf Labels (ESLs): These are physical devices with e-ink or LCDs, that emit information via wireless communication with the system.
* Central Server: We need a computer or server to run the backend application and database.

**Operating Systems**

* Server: Windows Server 2016+, Linux (Ubuntu 18.04 and later)
* ESLs: Work on embedded firmware that matches the hardware specs.

**Software Dependencies**

* Database: MySQL 5.7+ or PostgreSQL 10+, for data storage.
* Web server: Apache or Nginx for hosting the management interface
* Programming Environment: This is the backend, which backs up in Java, Python, or Node. js.

**Network Requirements**

* Wireless Networking: ESLs require Wi-Fi (802.11n/g/b) or Bluetooth Low Energy for server connectivity.
* Internet Access: Required when performing remote updates or executing compliance checks if necessary.

**Integration Compatibility**

* Integrations: Integrates with your existing inventory and point-of-sale (POS) systems.
* APIs: Provides RESTful APIs that facilitate seamless data exchange with other software applications.

## Design and Implementation Constraints

Creating the Smart Label System does have some real-world challenges to navigate around:

**Hardware Limitations**: Electronic Shelf Labels (ESLs) have limited memory and processing power, and thus firmware and software need to be extremely efficient and optimized.

**Regulatory Compliance**: The system must comply with strict retail pricing laws, and data protection laws, and compliance is a giant shape of the design.

**Integration Requirements**: It should be able to integrate with current inventory and POS systems, which means it has to be compatible with the existing technologies.

**Security Considerations**: We are dealing with sensitive pricing data here, so secure data transmission and storage are requisite.

**Communication Protocols**: There’s a reliance on wireless communication (such as Wi-Fi or Bluetooth), but store environments may introduce interference with signals — something to keep an eye on.

**Development Tools**: There are only so many options to operate on given—project or client requirements already dictate certain languages, databases, and frameworks that AMPs must use.

We specify the requirements for the system based on these constraints, which influences how the system will be built and whether it would work well in a retail environment.

## User Documentation

The Smart Label System is accompanied by a complete toolkit of not-so-smart & user-friendly documentation.

* **Quick Start Guide**: A brief, straight-to-the-point guide intended to get users up and running immediately.
* **Online Help:** Built-in, context-sensitive help available directly within the application for quick answers.
* **Tutorials:** Step-by-step walkthroughs that show users how to use the system’s main features.
* **Frequently Asked Questions (FAQ):** A go-to list of answers for common questions and troubleshooting tips.

## Assumptions and Dependencies

**Assumptions**

Here’s what we’re hoping for, in order for the Smart Label System to function as envisioned:

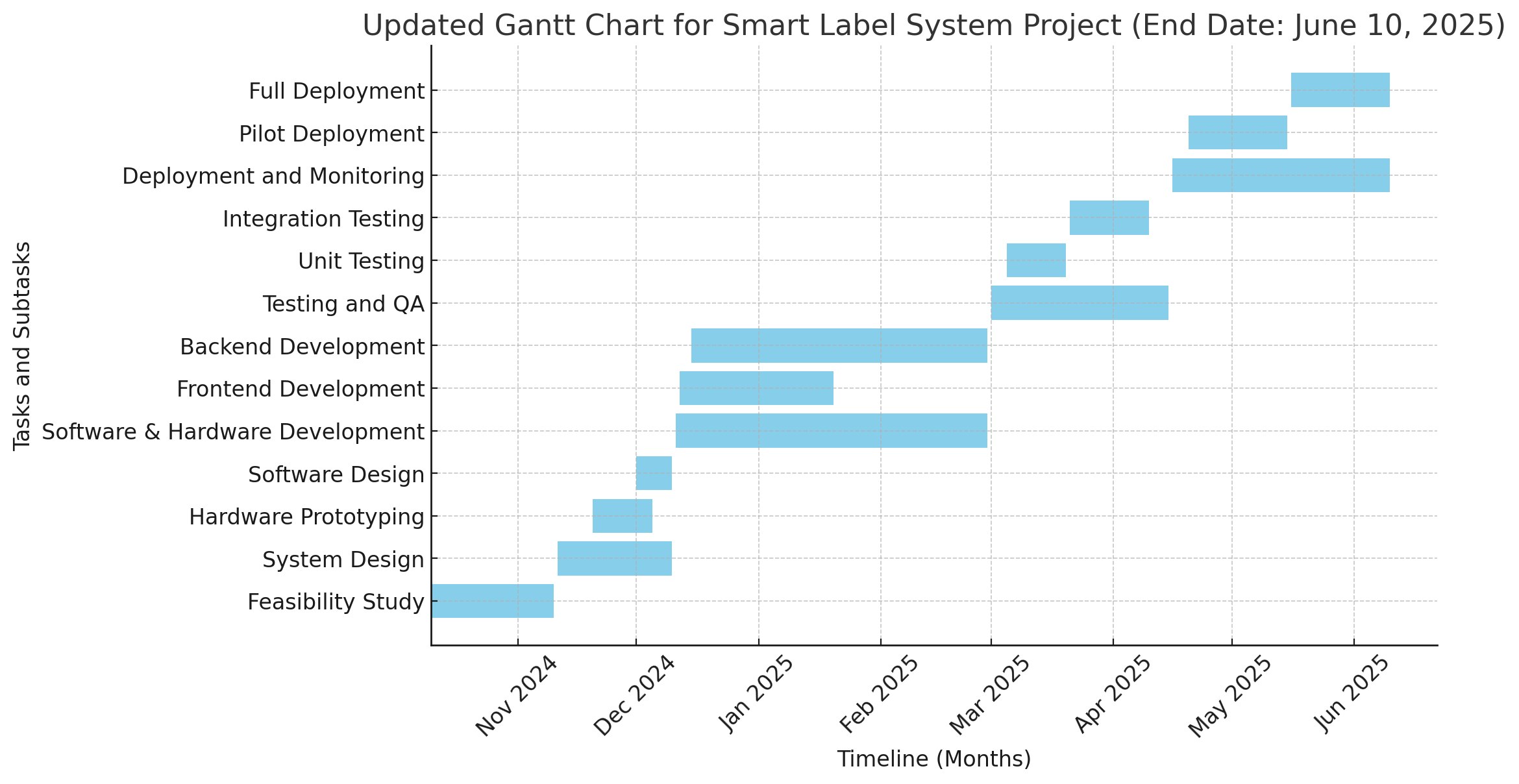
* Reliable Network Infrastructure: The store’s wireless infrastructure is established enough to manage communications between the system with the ESLs (electronic shelf labels).
* Touching From Barcodes to the ESL Base StationThe ESL device most commonly employed in direct store applicationsDifferential AdvantagesTechnicalAdvantagesHardware Compatibility The ESL devices are 100% compliant with communication protocols and technical requirements of the system.
* Integration Friendly: Existing inventory and POS systems are seamlessly integrated into the Smart Label System either via APIs or standard data exchange formats.
* Regulatory Stability: There won’t be changes in pricing laws while the system is developed or rolled out, promoting seamless implementation.
* User Competency: Store staff will undergo appropriate training, allowing them to utilize the system efficiently.

**Dependencies**

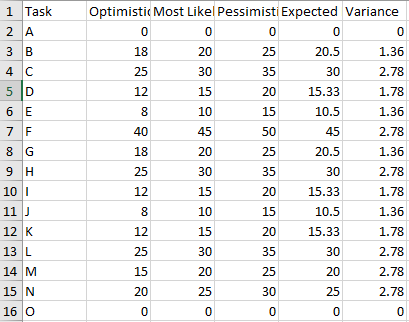
Below are the external dependencies that the system relies on for predictable results:

* Dependency on Third party hardware: The solution is based on electronic shelf labels and similar hardware parts which need to be manufactured by the production system.
* Software components: External libraries and frameworks that the service relies on for features (secure communication protocols, etc).
* Network services: The system requires reliable network access from service providers to verify updates and validate compliance with regulations by accessing regulatory services remotely.
* Maintenance and Support: You will need ongoing hardware and software support to ensure that your system runs smoothly and is always up-to-date.

# Gantt Chart Analysis



Gantt Chart



PERT Analysis

# External Interface Requirements

## User Interfaces

### Management Dashboard

* Purpose: Web based interface for Store Managers and Price Staff to perform pricing functions and also view system state.
* Features: Menu buttons and action buttons like “Update Price” and “Add Promotion.”
* Design Standards: Standard colour schemes, fonts, clean and consistent design using familiar GUI conventions.
* Error Handling: We display clear error prompts next to the field immediately when you enter wrong data so that you are able to correct it easily.
* **Electronic Shelf Labels (ESLs)**
* **What They Do:** Digital price tags displayed on store shelves, showing updated product prices and information.
* **Display:** High-contrast screens and legible fonts make it easy for customers to read prices at a glance.
* **Automatic Updates:** Prices are refreshed instantly and automatically—no manual updates needed.
* **Accessibility Features**
* **Responsive Design:** All interfaces adapt seamlessly to different screen sizes, whether you’re on a desktop, tablet, or mobile device.
* **Localization:** Multiple language options are available, so users can interact with the system in their preferred language.

## Hardware Interfaces

## Electronic Shelf Labels (ESLs)

* These e-ink or LCD devices are equipped with wireless communication (Wi-Fi or Bluetooth).
* The system sends updated pricing data directly to the ESLs, which display accurate, real-time product information on store shelves.

**2. Central Server Hardware**

* This is the powerhouse running the backend software.
* The server manages data storage, system operations, and handles tasks like processing price updates and managing inventory.
* It requires adequate processing power, memory, and storage to keep everything running smoothly.

**3.Point-of-Sale (POS) Systems**

* The system links to existing POS hardware to update pricing and inventory data.
* This reason is to make sure all information represented on ESLs, the dashboard and checkout is always consistent.

**4.Network Infrastructure**

* Between server and ESLs, communication allowed via routers and wireless access point.
* A stable and reliable network is necessary for real-time data transmission and efficiency of all systems.

## Software Interfaces

## Software Components

Operating Systems:

* The backend software runs on Windows Server 2016+ and Linux distros including Ubuntu 18.04+, giving retailers a choice in instructions and the location of the SSP deployment.

**Database Systems:**

* The system stores critical data such as product information, prices, user accounts, and audit logs using a relational database like MySQL 5.7+ or PostgreSQL 10+. Communication is through standard SQL queries.

**Web Servers:**

* Example it is a web-based dashboard powered by web servers such as Apache 2.4+ or Nginx 1.14+. It mimics object storage with much better ease of use.

**APIs and Integration:**

* RESTful APIs can easily integrate with inventory management, and point-of-sale (POS) systems. They take care of syncing product updates, stock levels, or pushing price changes via these APIs.

Communication Protocols:

* Regular books--HTTP/HTTPS requests were used for such things as web interactions and real time protocols such as MQTT or websockets were used to push updates to the electronic shelf labels.

**Data Flow**

**Incoming Data:**

* Price and Product Updates: Automatically sourced from an inventory or POS system, or manually via the management dashboard.
* Input: Store managers or pricing staff enter changes as part of promotional activities, price updates, etc.

**Outgoing Data:**

* Price Information: Sent directly to the electronic shelf labels to update product prices in real-time
* Alerts & Notifications: Sent to users, such as if prices violate legal constraints or for other significant events

**Shared Data:**

* Product Catalog: A central source maintained in the database that both the dashboard and ESLs pull from to ensure consistency.
* **User Credentials:** The shared authentication information throughout the system, allowing it to authorize who can log in and who cannot.
* **Key Implementation Constraints**

Security:

* To comply with data protection laws and shield sensitive pricing data, all data—both in motion and at rest—needs to be encrypted.

Compatibility:

* The software should integrate easily with existing systems (like POS and inventory management) without any major changes.

Performance:

* Because it need up-to-the-second information, it relies on efficient communication methods and optimized software for minimal latency.

## Communications Interfaces

**Management Dashboard Access**

How You Connect: The dashboard is served over secure HTTPS connections to the store managers and pricing staff, who access it from web browsers. This ensures sensitive information remains secure while being transmitted.

**Wireless ESLS Communication with ESLs**

Use of Protocols: The server connects through Wi-Fi (IEEE 802.11) or Bluetooth Low Energy (BLE) to ESLs, depending on the configuration.

Real-Time Updates The system continuously send data to the ESLs without latency using the MQTT or WebSocket protocol.

**Data Formats and APIs**

Data Consistency: Any data exchanged between components (e.g., product prices and updates) is formatted in JSON, making it easy to manage between each API.

Client-Server Interactions: The dashboard and server communicated through standard protocols such as HTTP/HTTPS.

**Security Measures**

Encryption: All communications are encrypted with SSL/TLS, which keeps sensitive pricing data safe.

Authentication Tokens: APIs are secured with authentication tokens that allow only authorized users and systems to reach them.

Email notifications: Alerts and notifications (such as when price limits are surpassed) use secure SMTP with TLS for sending.

**Why It Matters**

These communication interfaces ensure effective and secure operation of the system, allowing for real-time price updates in ESLs and seamless data transfer between the server, ESLs, and management dashboard. Every process, whether sending pricing data out to shelves or alerting users, is built with-speed secure and efficiency.

# System Features

The functional requirements for Smart Label System are detailed in this section under the main features of each component. Each feature consists of a description, priority, stimulus/response sequences, and detailed functional requirements.

## Real-Time Price Updates

### Description and Priority

This functionality ensures that the system can update product prices in real time from a single database.

**Priority**: High

### Stimulus/Response Sequences

• Stimulus: Retailer enters a change of price in the centralized database.

• Response: The corresponding ESL is updated in real-time.

### Functional Requirements

• **REQ-1**: The system shall retrieve price updates from the central database

• **REQ-2**: The system shall display updated prices on the ESL shortly after database changes.

• **REQ-3**: The system shall log all price changes with timestamps in a centralized record.

• **REQ-4**: The system shall notify staff in case of synchronization failures.

## Comprehensive Product Information

### Description and Priority

This feature provides customers with detailed product information via ESL displays.

**Priority**: Medium

### Stimulus/Response Sequences

• **Stimulus**: Retailer inputs product details into the database.

• **Response**: The ESL displays product name, description, nutritional facts, and promotional offers.

### Functional Requirements

• Initially,The generated requirements are numbered from REQ-1 to REQ-n as mentioned below:• REQ-1: The system shall show product name, description, facts, and promotion on ESL.

•REQ-2: The system should allow text, icons, and QR codes on ESLs.

• REQ-3: E = The system must maintain data integrity by comparing the information shown with the information stored in the database.

## Threshold Alerts

### Description and Priority

Other features include sending alerts when prices surpass regulatory limits or when products near expiration.

**Priority**: High

### Stimulus/Response Sequences

• Stimulus: Product price is above some predetermined threshold or product is close to expiration

• Alert: Staff and/or government officials are alerted to the presence of a criminal.

### Functional Requirements

• REQ-1: Users shall be able to set price and expiration alerts.

• REQ-2: The system must produce and dispatch alerts whenever marks are exceeded.

• REQ-3: System shall log all alerts for compliance reporting

## Customer Engagement Features

### Description and Priority

This feature provides customers with interactive tools for enhanced shopping experiences.

**Priority**: Medium

### Stimulus/Response Sequences

•1Scan the QR code in ESL4Stimulus: Customer scans the QR code displayed in ESL.

• Action: The system sends the customer to a page for that product or any special offer related.

### Functional Requirements

• REQ-1: System must produce QR codes for individual product.

• REQ-1: The system should connect QR codes to relevant product pages or promotional offers.

• REQ-3 The system will keep records of QR code scans for analytics.

## Data Analytics and Reporting

### Description and Priority

This feature provides insights into sales trends, customer preferences, and inventory levels.

**Priority**: Low

### Stimulus/Response Sequences

•Drive: Retailer makes analytics requests from the platform.

• Action: The system creates and sends the report.

### Functional Requirements

•REQ-1: System shall collect and store sales and inventory data.

• REQ-2: System must provide customizable reporting for sales trends and customer behavior.

• REQ-3: The system should support data export in standard formats (e.g., CSV, PDF).

## Energy Efficiency

### Description and Priority

This feature minimizes the energy consumption of ESLs to reduce operational costs.

**Priority**: High

### Stimulus/Response Sequences

•Stimulus: ESLs function at nominal or rest states.

• Response: The system uses low energy technologies to help serve within power usage limits.

### Functional Requirements

•REQ-1: ESLs should be implemented with e-ink displays to reduce power consumption.

• REQ-2: The ESLs must go into low-power mode when not in use.

• REQ-3: The once in a while shall be expected to monitor and provide ESL power usage

## Integration with Store Systems

### Description and Priority

This feature ensures seamless integration with existing inventory and point-of-sale systems.

**Priority**: High

### Stimulus/Response Sequences

• Stimulus: Inventory system is updated with price or stock information

• Respomce: These updates are immediately reflected in ESL.

### Functional Requirements

•REQ-1: The system should be able to call into existing inventory/pos systems using APIs

• REQ-2: Update must be in sync across all integrated platforms.

• REQ-3 The system shall log all integration transactions (for auditing)

## Compliance with Regulatory Standards

**Description and Priority**

This feature ensures adherence to retail and data privacy regulations.

**Priority**: High

**Stimulus/Response Sequences**

• **Stimulus**: New regulations are introduced or existing regulations are updated.

• **Response**: The system updates its compliance mechanisms accordingly.

**Functional Requirements**

•REQ-1: The system shall allow compliance with local pricing regulations.

• REQ-2: The system must encrypt sensitive information for privacy.

• REQ-3: System shall produce compliance audit reports.

# Other Nonfunctional Requirements

This section outlines nonfunctional requirements for the Smart Label System, focusing on attributes that enhance the system’s usability, scalability, and compliance.

## Scalability

• The system must support operations for supermarkets of varying sizes, from small grocery stores to large hypermarkets.

• ESL hardware and software should be modular to facilitate scaling without significant redesign or redevelopment.

## Usability

• The interface for both customers and administrators must be user-friendly and intuitive.

• Administrative interfaces should include tutorials and on-screen guidance for non-technical users.

• ESLs must provide clear, legible information with adjustable font sizes for accessibility.

## Availability

• The system must achieve a 99.9% uptime to ensure continuous operation during business hours.

• It should include fallback mechanisms to retain critical functionality during network outages.

## Compatibility

• The system must integrate seamlessly with existing inventory management and point-of-sale systems used in retail environments.

• It must support APIs for data exchange with external systems.

## Performance Requirements

### Response Time

• ESL updates must propagate across the entire store within **2 seconds** after a price change in the centralized database.

• System notifications for regulatory breaches must be delivered to administrators or regulatory bodies within **1 second** of detection.

### Data Throughput

• The system must handle a minimum of **500 concurrent ESL updates** without performance degradation.

• Data synchronization between the database and ESLs should support **10,000 transactions per hour**.

### Energy Efficiency

• ESL devices must consume less than **1mW** during standby mode and less than **10mW** during active updates.

• Battery life for ESL devices must last at least **5 years** under standard usage conditions.

### Real-Time Analytics

• The system must process and display sales and inventory analytics within **5 seconds** of a request.

## Safety Requirements

### Hardware Safety

• ESL hardware must comply with ISO 9001 standards to ensure durability and reliability.

• The design must prevent overheating, even under prolonged operation.

### Data Accuracy

• The system must ensure accurate data display on ESLs to avoid misleading customers, with validation mechanisms for price and product data.

### Regulatory Compliance

• The system must adhere to TRNC retail regulations, including price transparency standards.

• Alerts should notify staff immediately when pricing breaches regulatory limits, mitigating the risk of fines.

### Customer Safety

• ESL hardware should use non-toxic, eco-friendly materials for compliance with safety and environmental standards.

• Displays must be free of hazardous sharp edges or materials.

## Security Requirements

### Data Protection

• All customer and pricing data must be encrypted using **AES-256** encryption standards during storage and transmission.

• The system must comply with TRNC data protection laws and international standards, such as GDPR where applicable.

### Authentication

• The administrative interface must support multi-factor authentication (MFA) for secure access.

• User roles must define granular permissions, ensuring that only authorized personnel can modify price or product data.

### Intrusion Detection

• The system must include intrusion detection mechanisms to identify and respond to unauthorized access attempts.

• All access logs must be maintained for a minimum of **6 months** for auditing purposes.

### Network Security

• Communication between ESLs and the central database must occur over secure protocols (e.g., HTTPS, TLS 1.3).

• The system must prevent man-in-the-middle attacks through mutual authentication.

## Software Quality Attributes

### Reliability

• The system should demonstrate an error rate of less than **0.01%** during routine operations.

• Backup mechanisms must ensure no data loss during power failures or system crashes.

### Maintainability

• Codebases must follow modular programming principles to simplify debugging and updates.

• Comprehensive documentation for hardware and software components must be provided.

### Interoperability

• The system should support interoperability with third-party applications and devices.

• APIs must adhere to industry standards for smooth integration.

### Usability

• ESL displays must ensure legibility under varying lighting conditions.

• Administrative tools should provide predictive text and dropdown options to reduce manual input errors.

### Testability

• Automated test scripts must cover **90% of the system’s functionality** before deployment.

• Unit, integration, and stress testing should be conducted periodically.

### Portability

• The software must be deployable across multiple platforms, including cloud and on-premises systems.

• ESL devices should support interoperability across different network infrastructures (e.g., Wi-Fi, Bluetooth).

## Business Rules

### Role-Based Access Control

• Only designated administrators may modify product prices.

• Regulatory bodies must have read-only access to compliance-related data.

### Compliance with Retail Policies

• Prices must remain visible to customers at all times during store operating hours.

• Any discrepancy between ESL prices and database records must trigger an immediate alert.

### Customer Interaction

• Customers must be able to access additional product information through QR codes displayed on ESLs.

• Promotional details displayed on ESLs must align with store-approved offers.

### Inventory Accuracy

• Product information on ESLs must always match the inventory records.

• Stock level alerts must notify staff when items fall below reorder thresholds.

### Analytics and Reporting

• Sales data collected through the system must be anonymized to protect customer privacy.

• Reports generated must conform to TRNC tax and financial reporting standards.

# Other Requirements

## Database Design

This section consists of the database design. Database prepared in DBML(Database Management Language).

Below figure shows the database diagram.metin, ekran görüntüsü, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

The database written in DBML is also included below:

Table Products {

ProductID int [pk, increment, not null]

Name varchar(255) [not null]

Description text

NutritionalFacts text

CategoryID int [ref: > ProductCategories.CategoryID, not null]

Price decimal(10,2) [not null]

ExpiryDate date

StockQuantity int [default: 0, not null]

Indexes {

(Name)

}

}

Table ProductCategories {

CategoryID int [pk, increment, not null]

CategoryName varchar(255) [not null, unique]

}

Table PriceChanges {

ChangeID int [pk, increment, not null]

ProductID int [ref: > Products.ProductID, not null]

OldPrice decimal(10,2) [not null]

NewPrice decimal(10,2) [not null]

ChangeDateTime datetime [not null]

ChangedByUserID int [ref: > Users.UserID, not null]

}

Table Users {

UserID int [pk, increment, not null]

UserName varchar(255) [not null, unique]

Password varchar(255) [not null]

Role enum('Admin', 'Staff', 'Regulatory') [not null]

Email varchar(255) [not null, unique]

}

Table RegulatoryLimits {

LimitID int [pk, increment, not null]

ProductCategoryID int [ref: > ProductCategories.CategoryID, not null]

MaxPrice decimal(10,2) [not null]

RegulationDate date [not null]

}

Table Alerts {

AlertID int [pk, increment, not null]

ProductID int [ref: > Products.ProductID, not null]

AlertType enum('PriceExceeded', 'NearExpiry') [not null]

AlertDateTime datetime [not null]

Status enum('Pending', 'Resolved') [not null, default: 'Pending']

}

Table Sales {

SaleID int [pk, increment, not null]

ProductID int [ref: > Products.ProductID, not null]

Quantity int [not null]

SaleDateTime datetime [not null]

}

Table Reports {

ReportID int [pk, increment, not null]

GeneratedByUserID int [ref: > Users.UserID, not null]

ReportType enum('PricingHistory', 'Compliance', 'SalesTrend') [not null]

GeneratedDateTime datetime [not null]

FilePath varchar(255) [not null]

}

# Appendix A: Glossary

A

• API (Application Programming Interface): A set of protocols and tools that allow different software components to communicate with each other.

• Audit Logs: A record of all system events and user actions, used for monitoring and compliance purposes.

C

• Central Database: A centralized system that stores all product information, pricing data, and configuration settings for the Smart Label System.

• Compliance: Adherence to regulatory standards, such as TRNC retail laws and international data protection standards.

D

• Data Encryption: A process of converting data into a secure format to prevent unauthorized access.

• Data Synchronization: Ensuring data consistency between the central database and all connected devices, such as ESLs.

E

• Electronic Shelf Labels (ESLs): Digital price tags that display product information and pricing on retail shelves, updated in real time.

• E-Ink Display: A low-power display technology used in ESLs, providing high contrast and legibility.

I

• IoT (Internet of Things): A network of interconnected devices capable of sharing and receiving data, such as ESLs and central servers.

• Integration: The seamless connection between the Smart Label System and existing inventory and point-of-sale systems.

L

• Latency: The delay between initiating an action (e.g., updating a price) and observing its result (e.g., updated display on ESL).

• Localization: Adapting the system’s interface and content for different languages and regions.

M

• Management Dashboard: A web-based interface for administrators and staff to manage pricing, inventory, and system settings.

N

• Network Infrastructure: The underlying framework enabling wireless communication between ESLs, the server, and other system components.

P

• POS (Point of Sale): A system used for managing transactions in retail settings, often integrated with inventory management systems.

• Pricing Threshold: Predefined limits for product prices, ensuring they comply with regulations.

R

• Real-Time Updates: The capability of the system to immediately reflect changes, such as price updates or alerts.

• Regulatory Compliance: Adhering to legal requirements related to pricing and consumer protection laws.

S

• Scalability: The system’s ability to handle increasing amounts of work or expand to accommodate larger retail environments.

• Security Protocols: Established methods for protecting data transmission and storage, including HTTPS and TLS.

• Server: The central computer system that manages the database, processes updates, and communicates with ESLs.

• System Logs: A chronological record of operations and events within the system.

T

• Testability: The ease with which the system’s features and functionalities can be tested to ensure they meet requirements.

U

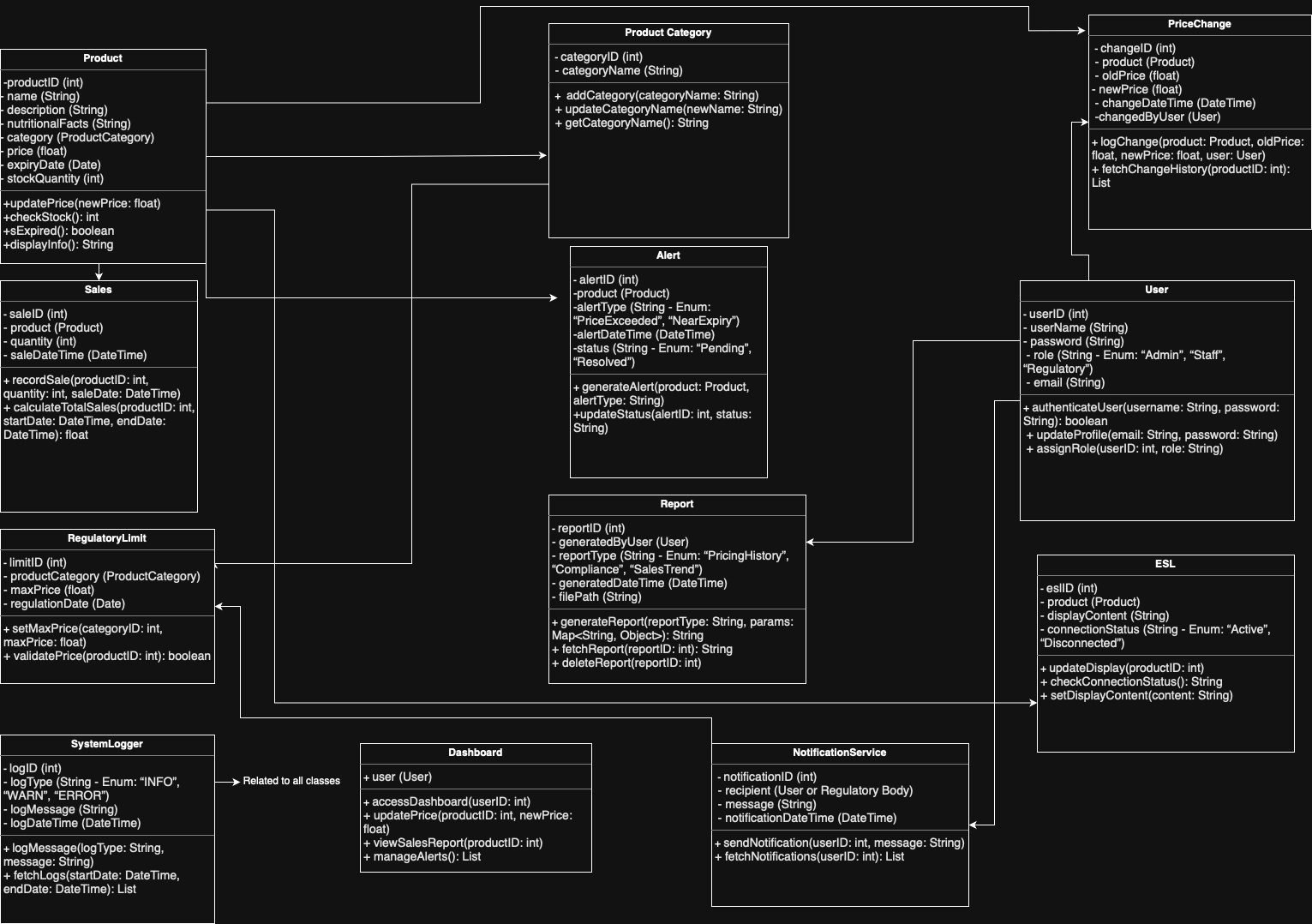
• User Roles: Different levels of system access based on job functions, such as administrators, pricing staff, and compliance officers.

• Usability: The ease with which users can learn and use the system effectively.

W

• Wireless Communication: The use of technologies like Wi-Fi and Bluetooth to transmit data between ESLs and the server.

# APPENDIX B : Relevant Diagrams

Class Diagram

metin, diyagram, ekran görüntüsü, çizgi içeren bir resim

Açıklama otomatik olarak oluşturulduBPMN Diagram

metin, diyagram, plan, teknik çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu

Context Diagram