Table of Contents

1. [Purpose 2](#_30j0zll)
   1. [Intended Audience 2](#_3znysh7)
   2. [Intended Use 2](#_tyjcwt)
   3. [Scope 2](#_1t3h5sf)
   4. [Definitions and Acronyms 2](#_2s8eyo1)
2. [Overall System Description 3](#_3rdcrjn)
   1. [Use Case Diagrams 3](#_lnxbz9)
   2. [System Architecture 4](#_1ksv4uv)
   3. [Functional Requirements 5](#_2jxsxqh)
      1. Main menu and Online Purchase [5](#_3j2qqm3)
      2. In Store Purchase7
      3. Payment Method9
      4. Additional features9

2.4 [Non-Functional Requirements 10](#_49x2ik5)

2.4.1 System responsiveness [10](#_147n2zr)

1. [Software Architecture 1](#_23ckvvd)0
   1. Overview [1](#_32hioqz)0
   2. User Interaction Layer………………………………………………………………………………………………………10
   3. Processing Layer……………………………………………………………………………………………………………….11
   4. Hardware Abstraction Layer……………………………………………………………………………………………..11

# Purpose

## Intended Audience

This SRS (Software Requirements Specification) document is designed for System and Software Engineers, Project Managers, and Retail Operations Managers who are tasked with developing and implementing the Supermarket Self-Checkout System. It provides comprehensive details necessary for all stages of software development and system deployment.

## Intended Use

The SRS outlines the overall system architecture, requirements, and software design for the Supermarket Self-Checkout System. This document serves as the basis for system configuration, software development, and ensures alignment with business requirements. It will be used to guide system testing, integration, and maintenance efforts, ensuring the checkout process is efficient and meets the specified user needs.

## Scope

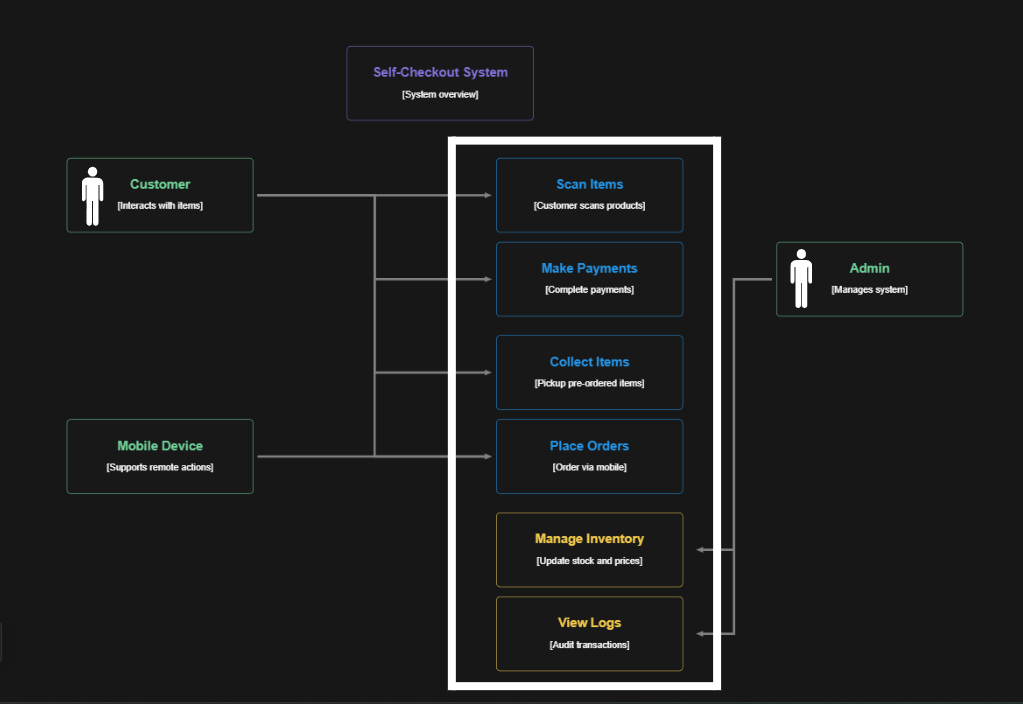
## The SRS outlines the overall system architecture, requirements, and software design for the Supermarket Self-Checkout System. This document serves as the basis for system configuration, software development, and ensures alignment with business requirements. It will be used to guide system testing, integration, and maintenance efforts, ensuring the checkout process is efficient and meets the specified user needs.

## Definitions and Acronyms

| **Acronym** | **Description** |
| --- | --- |
| IR | Infra Red |
| LED | Light Emitting Diode |
| NFC | Near Field Communication |
| SW | Software |
| HW | Hardware |
| DB | Database |

# Overall System Description

## Use Case Diagrams



## System Architecture

The System Architecture

#### Raspberry Pi Development Board

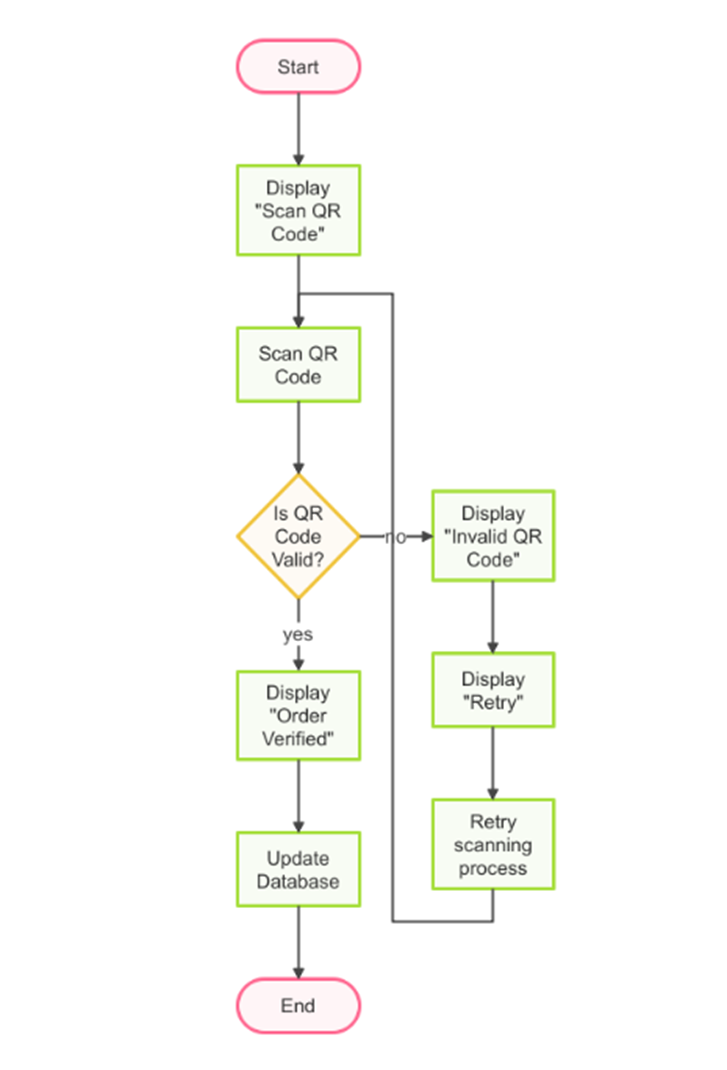




## Functional Requirements

### Main Menu and Online Purchase

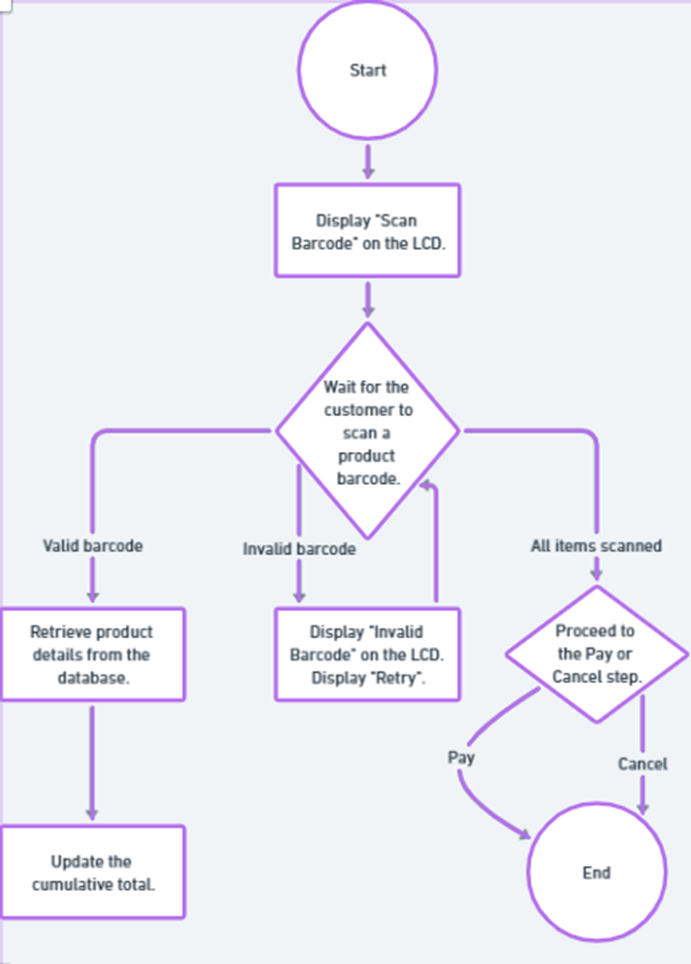
| **REQ\_ID** | **Requirement** |
| --- | --- |
| REQ-01 | When the system is first powered ON, the main menu shall be displayed on the LCD screen with the following options:   * **Line 1**: "1. Scan Items" * **Line 2**: "2. Online Order Pickup" |
| REQ-02 | When a customer selects **"Online Order Pickup"** from the main menu, the LCD shall display:   * **Line 1**: "Scan QR Code" * **Line 2**: "Processing..." |
| REQ-03 | Upon scanning a QR code, the system shall validate it against the database.   1. For a valid QR code, the LCD shall display:    * **Line 1**: "Order Verified"    * **Line 2**: "Prepare to collect" 2. For an invalid QR code, the LCD shall display:    * **Line 1**: "Invalid QR Code"    * **Line 2**: "Retry"   The system waits for another QR code scan |
| REQ-04 | For a valid QR code, the system shall update the database to mark the order as "Collected." |



**Figure 2**

2.3.2. In-Store Purchase

| **REQ\_ID** | **Requirement** |
| --- | --- |
| REQ-05 | When a customer selects **"Scan Items"** from the main menu, the LCD shall display:   * **Line 1**: "Scan barcode" * **Line 2**: "Total: $[Cumulative Total]" |
| REQ-06 | When a barcode is successfully scanned:   1. The system retrieves the product name and price from the database. 2. The LCD updates to display:    * **Line 1**: "[Product Name]"    * **Line 2**: "Price: $[Price]"   The cumulative total is updated and shown after 3 seconds. |
| REQ-07 | After scanning all items, the LCD shall display:   * **Line 1**: "1. Pay Now" * **Line 2**: "2. Cancel" |
| REQ-08 | If the customer selects **"Cancel"**, the LCD shall display:   * **Line 1**: "Transaction Cancelled" * **Line 2**: "Returning to main menu"   The system clears the scanned items and resets. |



**Figure 2**

### 2.3.3. Payment Method

### 

| **REQ\_ID** | **Requirement** |
| --- | --- |
| REQ-09 | If the customer selects "Pay Now", the system shall prompt for payment with the following options:   * Line 1: "1. PayWave" * Line 2: "2. PIN Entry" |
| REQ-10 | For PayWave payment:   1. The RFID reader shall detect the card. 2. The LCD shall display:    * Line 1: "Processing..."    * Line 2: "Do not remove card" 3. Upon success, the LCD shall display:    * Line 1: "Payment Successful"    * Line 2: "Thank you! |
| REQ-11 | For PIN-based payment:   1. The system shall prompt the customer to enter their PIN on the keypad. 2. The LCD shall display:    * Line 1: "Enter PIN"    * Line 2: "Press OK to confirm" 3. Upon success, the LCD shall display:    * Line 1: "Payment Successful"    * Line 2: "Thank you!" |

2.3.4. Additional Features

| **REQ\_ID** | **Requirement** |
| --- | --- |
| REQ-18 | If an invalid barcode is scanned, the LCD shall display:   * **Line 1**: "Invalid Barcode" * **Line 2**: "Please try again" |
| REQ-19 | The system shall detect tampering (e.g., multiple failed QR code or payment attempts) and activate the buzzer to alert staff. |

#### 2.4. Non-Functional Requirements

#### 2.4.1. System Responsiveness

#### The supermarket self-checkout system ensures that the user experience remains smooth and efficient under varying load conditions. This is achieved by monitoring system components and optimizing performance in real-time.

| REQ\_ID | Requirement |
| --- | --- |
| REQ-24 | System Startup: The system must initialize and display the main menu on the LCD screen within 5 seconds of power-on. |
| REQ-25 | Input Responsiveness: Keypad inputs and barcode scans must be processed within 100 milliseconds of user interaction. |
| REQ-26 | Error Handling: Any system error (e.g., database connection failure, hardware malfunction) must be logged and displayed within 1 second of occurrence. |
| REQ-27 | Queue Management: The system must process up to 3 concurrent checkout sessions without performance degradation. |
| REQ-28 | | Graceful Recovery: The system must recover automatically from temporary hardware or network failures without user intervention. | | --- | |

#### 

#### 3. Software Architecture

#### 3.1. Overview

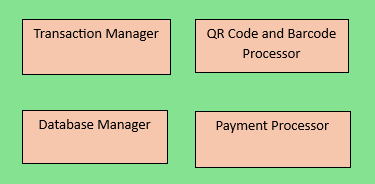
#### 

#### The software architecture of the supermarket self-checkout system is based on a modular design, ensuring scalability, maintainability, and efficient operation. The system is organized into three primary layers: User Interaction Layer, Processing Layer, and Hardware Abstraction Layer. These layers interact seamlessly to deliver a smooth checkout experience.

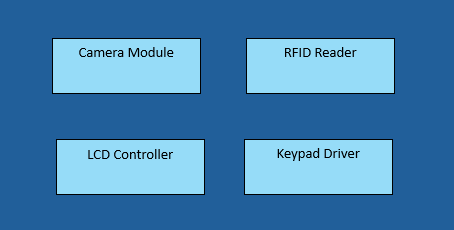
#### 3.2. User Interaction Layer

#### 

**3.3. Processing Layer**

****

**3.4. Hardware Abstraction Layer (HAL)**

****