Software Requirements Specification

Drop n Dash

This SRS document outlines all necessary features, modules, and requirements for the successful development and operation of the Drop n Dash platform. The document serves as a foundation for the design, development, and deployment phases.

Ali Sher Alwani
alisher.alwani@avancverasolution.com

Avancera Solution

Table of Contents

1. Introduction 1

2. Overall Description 2

3. System Features 3

4. External Interface Requirements 7

5. Non-functional Requirements 7

6. Other Requirements 8

7. Conclusion 8

1. Introduction

1.1 Purpose

The purpose of this SRS document is to outline the functional and non-functional requirements for the development of the Drop n Dash logistics service platform. This platform aims to facilitate the easy dispatch of parcels by leveraging a network of retailers, wholesalers, and third-party courier services, using QR codes for tracking and management.

1.2 Scope

Drop n Dash is a logistics platform that connects customers, retailers, wholesalers, and third-party courier services to streamline the process of sending parcels. The platform will include modules for QR code purchase and management, parcel tracking, user account management, and financial transactions.

1.3 Definitions, Acronyms, and Abbreviations

- **QR Code:** A machine-readable code used to track parcels.
- **API:** Application Programming Interface, used for integration with third-party courier services.
- **DHL, FedEx, TCS:** Examples of third-party courier services.

1.4 References

- **IEEE Std 830-1998:** IEEE Recommended Practice for Software Requirements Specifications.

1.5 Overview

This document is organized into sections detailing the overall description of the system, specific functional requirements, external interface requirements, system features, and non-functional requirements.

2. Overall Description

2.1 Product Perspective

Drop n Dash is a web and mobile application that will serve as a mediator between customers, retailers, wholesalers, and third-party couriers. The system will handle QR code generation and management, parcel tracking, and financial transactions.

2.2 Product Functions

- **Customer Module:** Allows customers to purchase QR codes, activate them, and track parcels.
- **Retailer Module:** Enables retailers to sell QR codes, activate them, and manage their wallet balances.
- **Wholesaler Module:** Allows wholesalers to sell QR code books, manage inventory, and handle financial transactions.
- **Admin Module:** Provides admins with tools to monitor all operations, manage user accounts, and oversee financial transactions.
- **Third-Party Courier Module:** Integrates with third-party couriers to facilitate parcel delivery.
  Note: The criteria will be to hit the third party APIs with all the parameter, and select the cheaper one.

2.3 User Classes and Characteristics

- **Customers:** General users with minimal technical knowledge, looking for easy and quick parcel sending solutions.
- **Retailers:** Store owners familiar with basic transaction processes, responsible for selling and activating QR codes.
- **Wholesalers:** Store owners or managers responsible for bulk transactions and inventory management.
- **Admins:** Users with advanced knowledge of the platform, responsible for overall system management.
- **Third-Party Couriers:** External service providers responsible for the final delivery of parcels.

2.4 Operating Environment

- **Web Application:** Compatible with major browsers like Chrome, Firefox, and Safari.
- **Mobile Application:** Compatible with iOS and Android devices.
- **Backend:** Cloud-based infrastructure with API integrations for third-party services.

2.5 Design and Implementation Constraints

- **Security:** Ensuring secure transactions and data privacy.
- **Scalability:** Handling a large volume of transactions and users.
- **Third-Party API Dependency:** Reliable integration with courier services.

2.6 User Documentation

- **User Manuals:** For customers, retailers, wholesalers, and admins.
- **API Documentation:** For third-party couriers.

2.7 Assumptions and Dependencies

- Reliable internet connectivity is required for all users.
- Third-party courier services have stable APIs for integration.
- Retailers and wholesalers maintain sufficient QR code inventory.

3. System Features

3.1 Customer Module

3.1.1 QR Code Purchase

- **Description:** Customers can purchase QR codes from nearby retailers or online.
- **Inputs:** Parcel size selection (Small, Medium, Large).
- **Outputs:** QR code receipt and tracking ID.
- **Functional Requirements:**
  - FR1: Allow customers to select parcel size.
  - FR2: Generate QR code upon purchase.
  - FR3: Term and Condition consent approval.
  - FR4: Provide payment options and confirm transactions.
  - FR5: Deliver the QR Code to the customer (Outside the application)

3.1.2 QR Code Activation

- **Description:** Retailer/Wholesaler activate QR codes for the customer by scanning the QR code and entering the customer details, on proceeding the customer receives a code via SMS on entering code the QR code get activated for the customer.
- **Inputs:** Activation code, customer details.
- **Outputs:** Activated QR code linked to parcel information.
- **Functional Requirements:**
  - FR1: Validate activation code.
  - FR2: Link QR code to customer and parcel details.
  - FR3: Deduct QR amount from Retailer wallet.

3.1.3 Parcel Drop-off and Tracking

- **Description:** Customers drop off parcels at any nearby retailer store for dispatch.
- **Inputs:** QR code, customer details, parcel details.
- **Outputs:** Tracking information sent via SMS/email.
- **Functional Requirements:**

- FR1: Allow customers to drop off parcels at any participating store.
- FR2: System will check the rates on the third party service provider APIs and will set the best option in term of (Delivery time, Price) for the deliver
- FR3: Enable real-time tracking of parcels.
- FR4: Send status updates via SMS and email.
- FR5: Add dispatch amount in participant store wallet.

3.2 Retailer Module

3.2.1 QR Code Sales and Activation

- **Description:** Retailers sell and activate QR codes for customers.
- **Inputs:** Parcel size, customer details.
- **Outputs:** Activated QR code and transaction receipt.
- **Functional Requirements:**
  - FR1: Manage QR code inventory.
  - FR2: Process QR code sales and generate receipts.
  - FR3: Activate QR codes and link them to customer parcels.

3.2.2 Wallet Management

- **Description:** Retailers manage their wallet for transactions related to QR code sales and parcel dispatch.
- **Inputs:** Sales transactions, QR code activation.
- **Outputs:** Updated wallet balance.
- **Functional Requirements:**
  - FR1: Track wallet balance based on QR code sales and activations.
  - FR2: Provide options to view transaction history.

3.2.3 Parcel Dispatch

- **Description:** Retailers manage the dispatch of parcels to third-party couriers.
- **Inputs:** Parcel details, QR code.
- **Outputs:** Parcel dispatched status.

- **Functional Requirements:**
  - FR1: Facilitate parcel handover to third-party couriers.
  - FR2: Update dispatch status in the system.
  - FR3: System will add the dispatched amount into the retailer wallet.

3.3 Wholesaler Module

3.3.1 QR Code Book Sales

- **Description:** Wholesalers sell QR code books to retailers.
- **Inputs:** Order details, payment information.
- **Outputs:** QR code book and transaction receipt.
- **Functional Requirements:**
  - FR1: Manage bulk QR code inventory.
  - FR2: Process wholesale orders and generate receipts.

3.3.2 Wallet

- **Description:** Wholesalers manage their wallet for transactions related to QR code book sales.
- **Inputs:** Sales transactions.
- **Outputs:** Updated wallet balance.
- **Functional Requirements:**
  - FR1: Track wallet balance based on sales of QR code books.
  - FR2: Provide options to view transaction history.

3.3.3 Claim QR and Packaging

- **Description:** Wholesalers can claim unused or faulty QR codes and packaging.
  Notes: Any QR which isn't scannable or shows the wrong parcel size example (Small, Medium, Large) other then what's printed on the parcel is faulty.
- **Inputs:** Claim details.
- **Outputs:** Adjusted inventory and wallet balance.
- **Functional Requirements:**
  - FR1: Allow wholesalers to submit claims for faulty or unused QR codes.
  - FR2: Adjust inventory and wallet balance based on approved claims.

3.4 Admin Module

3.4.1 User Management

- **Description:** Admins manage all user accounts including customers, retailers, and wholesalers.
- **Inputs:** User account details.
- **Outputs:** User account status (active, suspended, Rejected.).
- **Functional Requirements:**
  - FR1: View and manage all user accounts details.
  - FR2: Approve, reject, or suspend user accounts.

3.4.2 QR Code and Parcel Management

- **Description:** Admins oversee QR code distribution and parcel tracking across the platform.
- **Inputs:** QR code and parcel details.
- **Outputs:** Status reports and analytics.
- **Functional Requirements:**
  - FR1: Monitor QR code inventory and distribution.
  - FR2: Track parcel status from Activation to delivery.
  - FR3: Generate detailed reports on parcel delivery performance.

3.4.3 Financial Management

- **Description:** Admins oversee all financial transactions and wallet balances for retailers and wholesalers.
- **Inputs:** Transaction details, payment data.
- **Outputs:** Financial reports, reconciled balances.
- **Functional Requirements:**
  - FR1: Monitor financial transactions across the platform.
  - FR2: Reconcile wallet balances for retailers and wholesalers.
  - FR3: Generate financial reports.

3.4.4 System Configuration

- **Description:** Admins configure system settings including third-party courier integrations.
- **Inputs:** API keys, system settings.
- **Outputs:** Configured system parameters.
- **Functional Requirements:**
  - FR1: Configure API integrations with third-party courier services.

- FR2: Manage system-wide settings and configurations.

3.5 Third-Party Courier Module

3.5.1 Parcel Pickup and Delivery

- **Description:** Couriers receive parcel pickup requests and handle delivery to the final destination.
- **Inputs:** Pickup request, parcel details.
- **Outputs:** Delivery confirmation, status updates.
- **Functional Requirements:**
  - FR1: Integrate with the platform to receive pickup requests. - FR2: Update delivery status in real-time.
  - FR3: Provide delivery confirmation to the platform.

4. External Interface Requirements

4.1 User Interfaces

- **Web Interface:** User-friendly interface for customers, retailers, wholesalers, and admins.
- **Mobile Interface:** Simplified interface for on-the-go access.

4.2 Software Interfaces

- **Third-Party API Integration:** Integration with APIs from courier services for dispatch and tracking.
- **Payment Gateway Integration:** Secure integration with payment gateways for transactions.

4.3 Communications Interfaces

- **Email and SMS:** For notifications and status updates.
- **Push Notifications:** For real-time updates on parcel status.

5. Non-functional Requirements

5.1 Performance Requirements

- **Scalability:** The system must handle high volumes of transactions and users.
- **Response Time:** Actions like QR code activation and parcel tracking should occur in real-time.

5.2 Security Requirements

- **Data Protection:** All user and transaction data must be encrypted.

- **Access Control:** Role-based access for customers, retailers, wholesalers, and admins.

5.3 Reliability Requirements

- **Uptime:** The platform should maintain 99.9% uptime.
- **Fault Tolerance:** The system should recover gracefully from hardware or software failures.

5.4 Maintainability Requirements

- **Modularity:** The system should be modular for easy updates and maintenance.
- **Documentation:** Comprehensive documentation for developers and users.

5.5 Portability Requirements

- **Platform Independence:** The platform should work on various devices and operating systems.

6. Other Requirements

6.1 Legal and Regulatory Requirements

- **Compliance:** The platform must comply with local data protection regulations (e.g., GDPR).
- **Licensing:** Ensure all third-party integrations are appropriately licensed.

7. Conclusion

This Software Requirements Specification (SRS) document provides a comprehensive overview of the Drop n Dash logistics service platform, detailing its intended functionality, user roles, system features, and technical requirements. By clearly defining the user journeys and system components, this document serves as a critical guide for the development team to design, implement, and deploy the platform effectively.

The successful implementation of Drop n Dash will facilitate seamless logistics operations, empowering customers to easily send parcels, enabling retailers and wholesalers to manage QR code sales and dispatches efficiently, and integrating with third-party courier services to ensure reliable delivery. Adhering to the outlined requirements will help ensure that the platform meets business objectives, user expectations, and regulatory standards, ultimately delivering a robust and scalable solution for all stakeholders involved.

Moving forward, the development team should use this SRS as the foundation for the system's architecture and implementation. Regular reviews and updates to this document may be necessary as the project progresses and new requirements emerge.

The SRS concludes with a commitment to delivering a high-quality platform that is user-friendly, secure, and scalable, meeting the needs of all users and ensuring the long-term success of the Drop n Dash service.