

**Software Requirements Specification for Vinkay Market**

**CMU-SE 214 ACIS**

**Version 1.0**

Prepared by:  
**Group Name:** Vinkay Development Team

*Date: 08/09/2024*

**Instructor:** MSc. Nguyen Trung Thuan  
**Course:** Requirements Engineering  
**Lab Section:** CMU-SE 214 ACIS

**Team Member**:

1. Nguyen Quoc Vinh – 29219051305
2. Truong Ho Cong
3. Bui Van Duy
4. Chau Y Kazan
5. Tam Truong

**Document Create by:** NGUYEN QUOC VINH

**International School, Duy Tan University**

|  |  |  |  |
| --- | --- | --- | --- |
| **PROJECT INFORMATION** | | | |
| **Project Title** | PRODUCT BACKLOG DOCUMENT | | |
| **Start Date** | 08/09/2024 | **End Date** | 12/12/2024 |
| **Lead Institution** | International School, Duy Tan University | | |
| **Project Mentor** | Nguyen Trung Thuan, Msc. | | |
| **Partner Organization** | Duy Tan University | | |
| **Project Web URL** | https://bit.ly/trello\_cmu\_vinkay | | |
| **Team Members** | **Name** | **Email** | **Tel** |
| **Leader** | Nguyen Quoc Vinh | nguyenquocvinh.bocking@gmail.com | +84 888 6956 79 |
| **Member** | Truong Ho Cong | thcong2k5@gmail.com |  |
| Bui Van Duy | vanduy.24082005@gmail.com |  |
| Chau Y Kazan | tinzbinh748aacroll7@gmail.com |  |
| Tam Truong | Null |  |

Table of Contents

[2. Introduction 4](#_Toc184847405)

[1.1 DOCUMENT PURPOSE 4](#_Toc184847406)

[1.2 PRODUCT SCOPE 4](#_Toc184847407)

[1.3 INTENDED AUDIENCE AND DOCUMENT OVERVIEW 4](#_Toc184847408)

[1.4 DEFINITIONS, ACRONYMS AND ABBREVIATIONS 5](#_Toc184847409)

[1.5 DOCUMENT CONVENTIONS 5](#_Toc184847410)

[1.6 REFERENCES AND ACKNOWLEDGMENTS 5](#_Toc184847411)

[3. Overall Description 6](#_Toc184847412)

[2.1 PRODUCT OVERVIEW 6](#_Toc184847413)

[2.2 PRODUCT FUNCTIONALITY 7](#_Toc184847414)

[2.3 DESIGN AND IMPLEMENTATION CONSTRAINTS 8](#_Toc184847415)

[2.4 ASSUMPTIONS AND DEPENDENCIES 8](#_Toc184847416)

[3. Specific Requirements 10](#_Toc184847417)

[3.1 External Interface Requirements 10](#_Toc184847418)

[3.1.1 User Interfaces 10](#_Toc184847419)

[3.1. Functional Requirements 12](#_Toc184847420)

[3.2.1 User Authentication and Authorization 12](#_Toc184847421)

[3.3 Use Case Model 13](#_Toc184847422)

[3.3.1 Use Case: Purchase Product (U1) 13](#_Toc184847423)

[4. Other Non-functional Requirements 16](#_Toc184847424)

[4.1 Performance Requirements 16](#_Toc184847425)

[4.2 Safety and Security Requirements 17](#_Toc184847426)

[4.3 SOFTWARE QUALITY ATTRIBUTES 17](#_Toc184847427)

[5. Other Requirements 18](#_Toc184847428)

[Sprint 1: Project Initiation (September 7-20, 2024) 21](#_Toc184847429)

[1. Kickoff Meeting 21](#_Toc184847430)

[2. Requirements Gathering Session 21](#_Toc184847431)

[Sprint 2: Design Phase (September 21 - October 4, 2024) 22](#_Toc184847432)

[3. Architecture Design Meeting 22](#_Toc184847433)

[4. UI/UX Workshop 22](#_Toc184847434)

[Sprint 3: SRS Development (October 5-18, 2024) 22](#_Toc184847435)

[5. SRS Draft Review 22](#_Toc184847436)

[Sprint 4: Testing & Documentation (October 19 - November 1, 2024) 23](#_Toc184847437)

[6. QA Planning Session 23](#_Toc184847438)

[Sprint 5: Review & Refinement (November 2-15, 2024) 23](#_Toc184847439)

[7. Final SRS Review Meeting 23](#_Toc184847440)

[Sprint 6: Finalization (November 16-30, 2024) 23](#_Toc184847441)

[8. Document Finalization 23](#_Toc184847442)

[Final Submission Preparation (December 1-12, 2024) 23](#_Toc184847443)

[9. Final Team Meeting 23](#_Toc184847444)

[Team Effort Summary 24](#_Toc184847445)

[Tools Used 24](#_Toc184847446)

Revisions

| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| --- | --- | --- | --- |
| 0.1 | Team Lead | Initial draft and structure | 07/09/2024 |
| 0.2 | Backend Developer | Added functional requirements | 20/09/2024 |
| 0.3 | Frontend Developer | Added UI/UX requirements | 05/10/2024 |
| 0.4 | UI/UX Designer | Added mockups and user flow | 20/10/2024 |
| 0.5 | QA Engineer | Added non-functional requirements | 05/11/2024 |
| 0.6 | Full Team | Comprehensive review and updates | 20/11/2024 |
| 1.0 | Team Lead | Final version for submission | 11/12/2024 |

# Introduction

### 1.1 DOCUMENT PURPOSE

This Software Requirements Specification (SRS) document provides a comprehensive description of the Vinkay Mart mobile application. It details the functional and non-functional requirements, constraints, and specifications that will guide the development team in building the system. This document serves as a blueprint for the project, ensuring all stakeholders have a clear understanding of the application's objectives and features.

### 1.2 PRODUCT SCOPE

Vinkay Mart is a mobile e-commerce platform designed to connect local vendors with customers in Vietnam. The application aims to:

* Provide a user-friendly interface for browsing and purchasing products from local vendors
* Offer a secure and efficient payment system
* Enable vendors to manage their product listings and orders
* Implement a rating and review system for products and vendors
* Facilitate order tracking and delivery management
* Support multiple languages (Vietnamese and English)

The Vinkay Mart app will be available on both iOS and Android platforms, targeting smartphone users in Vietnam.

### 1.3 INTENDED AUDIENCE AND DOCUMENT OVERVIEW

**This document is intended for:**

* Development Team: To understand the technical requirements and implement the features
* Project Managers: To plan and track the project progress
* Quality Assurance Team: To develop test plans and ensure the application meets the specified requirements
* Stakeholders: To understand the scope and capabilities of the application

**The document is organized as follows:**

* Section 1 provides an introduction and overview of the project
* Section 2 describes the overall product, its features, and constraints
* Section 3 details the specific functional and interface requirements
* Section 4 outlines non-functional requirements such as performance and security
* Section 5 covers any additional requirements
* Appendices provide supplementary information including a data dictionary and group log

### 1.4 DEFINITIONS, ACRONYMS AND ABBREVIATIONS

* SRS: Software Requirements Specification
* UI: User Interface
* UX: User Experience
* API: Application Programming Interface
* CRUD: Create, Read, Update, Delete
* MVP: Minimum Viable Product
* QA: Quality Assurance
* VND: Vietnamese Dong (currency)

### 1.5 DOCUMENT CONVENTIONS

**This document follows these conventions:**

* Arial, Inter font, size 11 for body text
* Bold for section headings
* Italics for emphasis or technical terms when first introduced
* Numbered lists for sequential instructions
* Bullet points for non-sequential lists
* Code snippets in monospace font

### 1.6 REFERENCES AND ACKNOWLEDGMENTS

* IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications
* Figma Design Guidelines: <https://figma.com/design>
* iOS Human Interface Guidelines: <https://developer.apple.com/design/human-interface-guidelines/>

# Overall Description

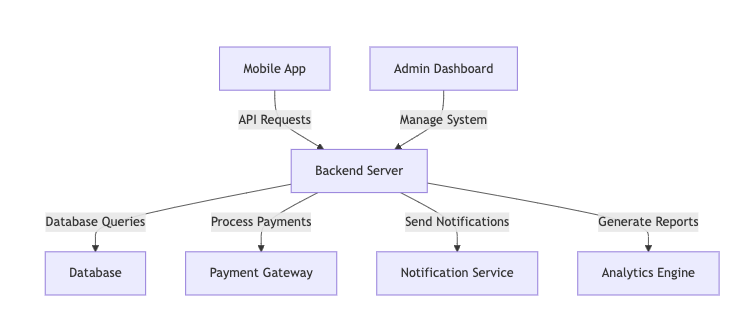
### 2.1 PRODUCT OVERVIEW

Vinkay Mart is a mobile e-commerce application designed to create a digital marketplace connecting local vendors in Vietnam with customers. The app provides a platform for small to medium-sized businesses to showcase their products and reach a wider audience, while offering customers a convenient way to discover and purchase local goods.

**Key components of the Vinkay Mart system include:**

1. **Mobile Application:**
   * Customer interface for browsing, purchasing, and managing orders
   * Vendor interface for managing products, orders, and store information
2. **Backend Server:**
   * API endpoints for mobile app communication
   * Database management
   * Authentication and authorization
3. **Payment Gateway:**
   * Integration with local payment providers
   * Secure transaction processing
4. **Notification System:**
   * Push notifications for order updates and promotions
   * In-app messaging between customers and vendors
5. **Analytics Dashboard:**
   * Sales and performance metrics for vendors
   * User behavior analysis for app improvement

The following diagram illustrates the high-level architecture of the Vinkay Mart system:



### 2.2 PRODUCT FUNCTIONALITY

The Vinkay Mart application must perform the following major functions:

• User Account Management

* Customer registration and authentication
* Vendor registration and verification
* Profile management and settings

**• Product Management**

* Product listing and catalog browsing
* Category and subcategory organization
* Search and filter functionality
* Inventory tracking

**• Order Processing**

* Shopping cart management
* Checkout process
* Payment processing
* Order tracking

**• Vendor Operations**

* Store profile management
* Product inventory management
* Order fulfillment
* Sales analytics

**• Customer Service**

* Review and rating system
* Customer support ticketing
* Chat functionality
* Notification system

### 2.3 DESIGN AND IMPLEMENTATION CONSTRAINTS

1. **Technical Constraints**
   * Mobile application must be developed using Flutter framework for cross-platform compatibility
   * Backend must be implemented using Node.js and Express.js
   * Database implementation must use MongoDB for product catalog and PostgreSQL for transactions
   * Must follow COMET (Component-based Model-driven Engineering Technology) method for software design
   * Must use UML 2.5 specification for all modeling diagrams
2. **Hardware Constraints**
   * Minimum supported Android version: 8.0 (API level 26)
   * Minimum supported iOS version: 13.0
   * Maximum app size: 50MB
   * Minimum 2GB RAM required on user devices
3. **Network Constraints**
   * Must function on 3G networks and above
   * Must handle intermittent connectivity
   * Maximum response time of 3 seconds for main operations
4. **Security Constraints**
   * Must comply with Vietnam's cybersecurity laws
   * Must implement SSL/TLS encryption for all data transmission
   * Must follow OWASP security guidelines
   * Must implement rate limiting for API endpoints
5. **Interface Constraints**
   * Must follow Material Design guidelines for Android
   * Must follow Human Interface Guidelines for iOS
   * Must support both Vietnamese and English languages
   * Must support right-to-left layouts for future language additions

### 2.4 ASSUMPTIONS AND DEPENDENCIES

**Assumptions:**

1. **Users have access to smartphones with:**
   * Stable internet connection
   * GPS capability
   * Camera functionality
   * Sufficient storage space (minimum 100MB)
2. **Market Conditions:**
   * Continued growth in e-commerce adoption in Vietnam
   * Stable regulatory environment for e-commerce
   * Available pool of local vendors willing to participate
3. **Technical Environment:**
   * Cloud services remain available and cost-effective
   * Payment gateways maintain their API compatibility
   * Mobile OS platforms maintain backward compatibility

**Dependencies:**

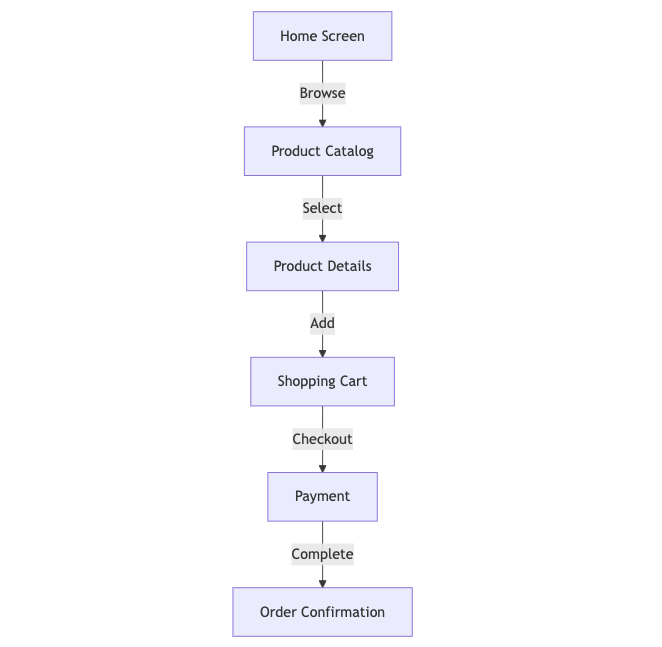
1. **External Services:**
   * Payment gateway services (VNPay, MoMo)
   * Cloud hosting provider (AWS/GCP)
   * Maps API service (Google Maps)
   * Push notification services (Firebase)
2. **Third-party Components:**
   * Flutter SDK and dependencies
   * Node.js runtime and packages
   * MongoDB and PostgreSQL databases
   * Analytics tools (Firebase Analytics)
3. **Development Tools:**
   * CI/CD pipeline tools
   * Testing frameworks
   * Code quality tools

# 3. Specific Requirements

## 3.1 External Interface Requirements

## 3.1.1 User Interfaces

The Vinkay Mart app will implement the following key interfaces:

1. Customer Mobile Interface:
2. Key Interface Elements:

* Bottom navigation bar for main sections
* Search bar with filters
* Product cards with images and basic info
* Shopping cart icon with item count
* User profile menu
* Order status tracker

1. Vendor Dashboard Interface:

* Product management console
* Order management system
* Analytics dashboard
* Inventory control panel
* Customer message center

1. Interaction Methods:

* Touch gestures (tap, swipe, pinch)
* Pull-to-refresh
* Infinite scrolling
* Form inputs
* Modal dialogs

#### 3.1.2 Hardware Interfaces

The application interfaces with the following hardware components:

* Device Sensors:
* Camera: Product image capture, QR code scanning
* GPS: Location services, delivery tracking
* Accelerometer: Screen orientation
* Biometric sensors: Secure authentication
* Network Hardware:
* WiFi adapter
* Cellular data module
* Bluetooth (for potential POS integration)
* Storage:
* Internal storage for app data
* Cache storage for images
* Secure element for payment info

#### 3.1.3 Software Interfaces

The application integrates with the following software systems:

* Backend Services:
* RESTful API endpoints
* WebSocket connections for real-time updates
* GraphQL API for complex queries
* External Services:
* Payment gateway APIs
* Cloud storage services
* Push notification services
* Analytics platforms
* Mobile OS Services:
* File system access
* Permission management
* Background services
* Local notifications

## Functional Requirements

## 3.2.1 User Authentication and Authorization

**F1**: User Registration

* The system shall allow users to register using email or phone number
* The system shall verify email/phone through OTP
* The system shall enforce password complexity requirements

**F2**: User Authentication

* The system shall support multiple login methods
* The system shall implement session management
* The system shall support password reset functionality

#### 3.2.2 Product Management

**F3**: Product Listing

* The system shall allow vendors to create and manage products
* The system shall support multiple product categories
* The system shall handle product variations

**F4**: Inventory Management

* The system shall track product stock levels
* The system shall notify vendors of low stock
* The system shall prevent overselling

## 3.3 Use Case Model

## 3.3.1 Use Case: Purchase Product (U1)

**Author**: Frontend Developer

**Purpose**: Enable customers to purchase products through the application

Requirements Traceability: F1, F2, F3, F4

**Priority**: High

**Preconditions:**

* User is authenticated
* Product is in stock
* User has valid payment method

**Post conditions:**

* Order is created
* Payment is processed
* Inventory is updated
* Confirmation is sent

**Actors:**

* Primary: Customer
* Secondary: Payment System, Inventory System

**Flow of Events:**

1. Basic Flow:
2. Customer browses products
3. Customer adds items to cart
4. Customer proceeds to checkout
5. Customer selects delivery address
6. Customer chooses payment method
7. System processes payment
8. System confirms order
9. Alternative Flow:
10. Payment declined
11. Item out of stock
12. Delivery address not supported
13. Exceptions:
14. Network error
15. System timeout
16. Payment gateway error

#### 3.3.2 Use Case: Manage Products (U2)

Author: Backend Developer

Purpose: Enable vendors to create, update, and manage their product listings and inventory on the Vinkay Mart platform

Requirements Traceability: F3, F4

Priority: High

Preconditions:

* Vendor is authenticated
* Vendor account is verified
* Vendor has necessary permissions

Post conditions:

* Product information is updated in database
* Inventory levels are adjusted
* Product changes are visible to customers
* Activity is logged for audit purposes

Actors:

* Primary: Vendor
* Secondary: Product Management System, Storage System

Flow of Events:

1. **Basic Flow:**
   * Vendor accesses product management dashboard
   * Vendor selects action (create/edit/delete product)
   * Vendor inputs/updates product information
   * Vendor sets inventory levels
   * Vendor uploads product images
   * System validates input
   * System saves changes
   * System confirms successful update
2. **Alternative Flow:**
   * Invalid product information
     + System displays validation errors
     + Vendor corrects information
     + System saves valid changes
   * Image upload fails
     + System notifies vendor
     + Vendor retries upload or skips
   * Insufficient storage space
     + System notifies vendor
     + Vendor deletes old images or upgrades storage
3. **Exceptions**:
   * Network connection lost
     + System saves draft
     + System restores session when connection returns
   * System maintenance
     + System notifies vendor
     + Changes are queued for processing
   * Database error
     + System logs error
     + System notifies admin
     + Vendor retries operation
4. **Includes:**

* Image Upload (U2.1)
* Inventory Update (U2.2)
* Price Management (U2.3)

Notes/Issues:

* Consider implementing bulk product upload feature
* Need to define image size and format restrictions
* Consider adding product variation management
* Plan for international pricing support

# Other Non-functional Requirements

## 4.1 Performance Requirements

P1: Response Time

* Home screen shall load within 3 seconds
* Product search shall return results within 2 seconds
* Payment processing shall complete within 5 seconds

P2: Scalability

* System shall support 10,000 concurrent users
* Database shall handle 1 million products
* File storage shall support 5TB of images

P3: Availability

* System uptime shall be 99.9%
* Planned maintenance shall be conducted during off-peak hours
* Recovery time objective (RTO) shall be 1 hour

## 4.2 Safety and Security Requirements

S1: Data Protection

* All personal data shall be encrypted at rest
* All network communications shall use SSL/TLS
* Payment information shall be tokenized

S2: Authentication Security

* Passwords shall be hashed using bcrypt
* Failed login attempts shall be rate-limited
* Sessions shall timeout after 30 minutes

S3: Transaction Security

* All financial transactions shall be atomic
* System shall maintain audit logs
* Fraud detection system shall be implemented

### 4.3 SOFTWARE QUALITY ATTRIBUTES

#### 4.3.1 Reliability

* Mean Time Between Failures (MTBF): 720 hours
* Automatic error recovery mechanisms
* Data backup every 6 hours

#### 4.3.2 Maintainability

* Modular architecture design
* Comprehensive API documentation
* Automated testing coverage > 80%

#### 4.3.3 Usability

* Maximum 3 clicks to complete main tasks
* Consistent UI patterns
* Clear error messages
* Help documentation available

# Other Requirements

1. **Internationalization**

* Support for multiple languages
* Localized currency formats
* Time zone handling

1. **Legal Compliance**

* GDPR compliance for future expansion
* Local e-commerce regulations
* Data privacy laws

1. **Documentation**

* API documentation
* User manuals
* System administration guides

Appendix A – Data Dictionary

| **Term** | **Description** | **Type** | **Constraints** |
| --- | --- | --- | --- |
| UserID | Unique identifier for users | UUID | 36 characters |
| ProductID | Unique identifier for products | UUID | 36 characters |
| OrderID | Unique identifier for orders | UUID | 36 characters |
| Price | Product price | Decimal | Positive number |
| Stock | Available quantity | Integer | Non-negative |
| OrderStatus | Current state of order | Enum | Created, Paid, Shipped, Delivered |
| UserRole | User's role in system | Enum | Customer, Vendor, Admin |
| PaymentStatus | Status of payment | Enum | Pending, Completed, Failed |

Appendix B - Group Log

## Sprint 1: Project Initiation (September 7-20, 2024)

### Kickoff Meeting

* **Date**: September 7, 2024
* **Duration**: 2 hours
* **Attendees**: All team members
* **Location**: Online (Google Meet)
* **Agenda**:
  1. Team introductions
  2. Project overview
  3. Role assignments
  4. Development methodology discussion
* **Decisions Made**:
  1. Selected Flutter for mobile development
  2. Chose MERN stack for backend
  3. Agreed on bi-weekly sprint schedule
* **Action Items**:
  1. Team Lead: Create project timeline
  2. UI/UX: Research competitor apps
  3. Backend: Setup development environment
  4. Frontend: Create initial wireframes
  5. QA: Draft testing strategy

### Requirements Gathering Session

* **Date**: September 15, 2024
* **Duration**: 3 hours
* **Attendees**: All team members + Stakeholders
* **Location**: Conference Room A
* **Topics Covered**:
  + Market analysis
  + User personas
  + Core features
  + Technical constraints
* **Deliverables**:
  + Initial feature list
  + Project constraints document
  + Risk assessment matrix

## Sprint 2: Design Phase (September 21 - October 4, 2024)

### Architecture Design Meeting

* **Date**: September 25, 2024
* **Duration**: 4 hours
* **Attendees**: Development team
* **Key Discussions**:
  + System architecture
  + Database design
  + API structure
  + Security measures
* **Outputs**:
  + Architecture diagrams
  + Database schema
  + API documentation draft

### UI/UX Workshop

* **Date**: October 2, 2024
* **Duration**: 6 hours
* **Activities**:
  + Wireframe review
  + User flow mapping
  + Design system creation
  + Prototype development

## Sprint 3: SRS Development (October 5-18, 2024)

### SRS Draft Review

* **Date**: October 12, 2024
* **Duration**: 3 hours
* **Review Points**:
  + Functional requirements
  + Non-functional requirements
  + Use cases
  + Interface specifications
* **Feedback Incorporated**:
  + Added performance metrics
  + Expanded security requirements
  + Updated user interfaces
  + Enhanced data dictionary

## Sprint 4: Testing & Documentation (October 19 - November 1, 2024)

### QA Planning Session

* **Date**: October 25, 2024
* **Duration**: 4 hours
* **Topics**:
  + Test strategy
  + Test cases
  + Automation framework
  + Performance testing plan

## Sprint 5: Review & Refinement (November 2-15, 2024)

### Final SRS Review Meeting

* **Date**: November 10, 2024
* **Duration**: 5 hours
* **Major Updates**:
  + Incorporated stakeholder feedback
  + Updated technical specifications
  + Refined use cases
  + Completed traceability matrix

## Sprint 6: Finalization (November 16-30, 2024)

### Document Finalization

* **Date**: November 25, 2024
* **Activities**:
  + Final proofreading
  + Format checking
  + Version control
  + Approval signatures

## Final Submission Preparation (December 1-12, 2024)

### Final Team Meeting

* **Date**: December 11, 2024
* **Duration**: 2 hours
* **Closing Activities**:
  + Document compilation
  + Final review
  + Submission preparation
  + Project retrospective

## Team Effort Summary

* Total Meetings: 15
* Total Hours: 45
* Document Versions: 6
* Individual Contributions:
  + Team Lead: Project management, document review
  + Frontend Developer: UI/UX specifications, user flows
  + Backend Developer: System architecture, API design
  + QA Engineer: Testing requirements, quality attributes
  + UI/UX Designer: Interface design, user experience

## Tools Used

* Project Management: Jira
* Documentation: Google Docs, Markdown
* Design: Figma
* Collaboration: Slack, Google Meet
* Version Control: Git, GitHub