Software Requirement Specification (SRS) for E-Commerce Web Application

1. Introduction

This document specifies the functional and non-functional requirements for an E-Commerce Web Application developed using Spring Boot and Thymeleaf.
The system will allow three types of users —
Admin, Merchant, and Customer — to interact with the platform based on role-specific permissions.

2. Scope

The E-Commerce Application aims to provide a simple and efficient way for merchants to showcase and manage products, for customers to shop, and for admins to manage the ecosystem.

Key highlights

Session-based authentication (HTTP Session).
Cloudinary integration for product image storage.
Razorpay integration for payment processing.
Admin module for order management and merchant product approvals.

3. Technology Stack

Technology	Description
Spring Boot	Backend Framework
Thymeleaf	Server-side Rendering Engine
Cloudinary	Image Storage Service
Razorpay	Payment Gateway
MySQL	Database (Assumed SQL-based)
HTML/CSS/Bootstrap	Frontend Styling

4. Modules Overview

Module	Description
Admin	Login, Approve/Reject Products, Manage Orders, View Overview
Merchant	Register, Login, Add/View/Update/Delete Products, Image Upload via Cloudinary
Customer	Register, Login, Browse Products, Add to Cart, Place Orders, Make Payments via Razorpay, Track Orders

5. Functional Requirements

5.1 Admin Module

Admin can log in using email and password.
Admin can approve or reject products.
Admin can view and manage customer orders.
Admin can access an overview dashboard (statistics of orders, merchants, customers).

5.2 Merchant Module

Merchants can register via email and password.
Merchants can log in and manage their profile.
Merchants can add new products with images (stored on Cloudinary).
Merchants can view, update, and delete their listed products.
Merchants can track their product approval status.

5.3 Customer Module

Customers can register via email and password.
Customers can log in and manage their profile.
Customers can browse products by category or search.
Customers can add products to cart.
Customers can place orders and complete payment via Razorpay.
Customers can view order history and track order status.

---

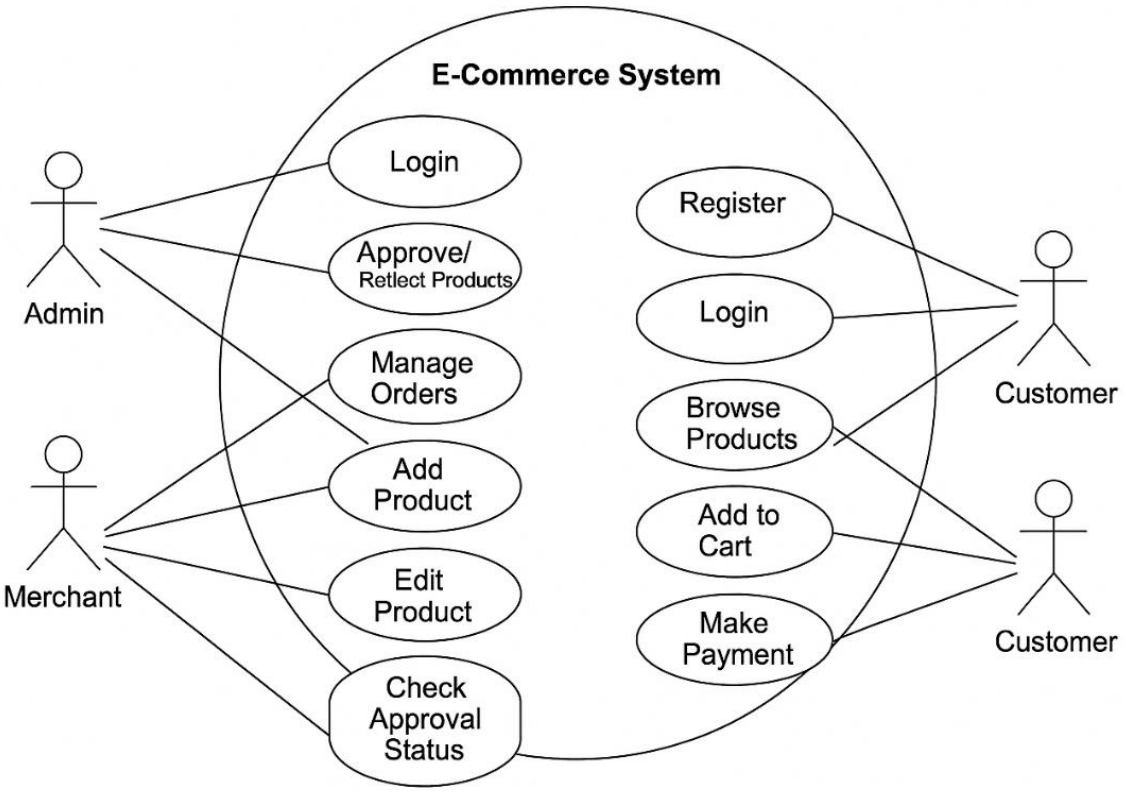
6. Non-Functional Requirements

- **Security:** Session-based authentication using HttpSession.
- **Performance:** The system should handle up to 500 concurrent users efficiently.
- **Scalability:** Cloudinary will handle image storage scalability.
- **Payment Safety:** Razorpay's secure transaction flow will be used.

---

7. System Flow Overview

1. Merchant registers → Logs in → Adds Products → Products sent for Admin Approval.
2. Admin approves/rejects products.
3. Customer registers → Logs in → Browses Products → Adds to Cart → Pays via Razorpay → Tracks Orders.



ADMIN Table

Field Name	Data Type	Constraints	Description
id	BIGINT	PRIMARY KEY, AUTO_INCREMENT	Unique ID (PK)
name	VARCHAR(100)	NOT NULL	Admin Name
email	VARCHAR(100)	UNIQUE, NOT NULL	Admin Email
password	VARCHAR(255)	NOT NULL	Hashed Password

MERCHANT Table

Field Name	Data Type	Constraints	Description
id	BIGINT	PRIMARY KEY, AUTO_INCREMENT	Unique ID (PK)
name	VARCHAR(100)	NOT NULL	Merchant Name
email	VARCHAR(100)	UNIQUE, NOT NULL	Merchant Email
password	VARCHAR(255)	NOT NULL	Hashed Password
registered_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Registration Date

PRODUCT Table

Field Name	Data Type	Constraints	Description
id	BIGINT	PRIMARY KEY, AUTO_INCREMENT	Unique ID (PK)
name	VARCHAR(150)	NOT NULL	Product Name
description	TEXT		Product Description
price	DECIMAL(10,2)	NOT NULL	Product Price
image_url	VARCHAR(255)	NOT NULL	Cloudinary Image URL
status	ENUM('PENDING', 'APPROVED', 'REJECTED')	DEFAULT 'PENDING'	Admin Approval Status
merchant_id	BIGINT	FOREIGN KEY REFERENCES merchant(id)	Linked Merchant
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Product Creation Date

CUSTOMER Table

Field Name	Data Type	Constraints	Description
id	BIGINT	PRIMARY KEY, AUTO_INCREMENT	Unique ID (PK)
name	VARCHAR(100)	NOT NULL	Customer Name
email	VARCHAR(100)	UNIQUE, NOT NULL	Customer Email
password	VARCHAR(255)	NOT NULL	Hashed Password
registered_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Registration Date

ORDER_ITEM Table

Field Name	Data Type	Constraints	Description
id	BIGINT	PRIMARY KEY, AUTO_INCREMENT	Unique ID (PK)
order_id	BIGINT	FOREIGN KEY REFERENCES orders(id)	Linked Order
product_id	BIGINT	FOREIGN KEY REFERENCES product(id)	Ordered Product
quantity	INT	NOT NULL	Quantity Purchased
price	DECIMAL(10,2)	NOT NULL	Unit Price at Time of Purchase

ORDERS Table

Field Name	Data Type	Constraints	Description
id	BIGINT	PRIMARY KEY, AUTO_INCREMENT	Unique Order ID (PK)
customer_id	BIGINT	FOREIGN KEY REFERENCES customer(id)	Linked Customer
total_amount	DECIMAL(10,2)	NOT NULL	Final Order Amount
payment_status	ENUM('PENDING', 'PAID', 'FAILED')	DEFAULT 'PENDING'	Payment Status
order_status	ENUM('PLACED', 'SHIPPED', 'DELIVERED', 'CANCELLED')	DEFAULT 'PLACED'	Order Lifecycle Status
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Order Creation Timestamp

PAYMENT Table

Field Name	Data Type	Constraints	Description
id	BIGINT	PRIMARY KEY, AUTO_INCREMENT	Unique Payment ID (PK)
order_id	BIGINT	FOREIGN KEY REFERENCES orders(id)	Linked Order
payment_id	VARCHAR(100)	UNIQUE, NOT NULL	Razorpay Payment Reference
amount	DECIMAL(10,2)	NOT NULL	Payment Amount
status	ENUM('SUCCESS', 'FAILED', 'PENDING')	DEFAULT 'PENDING'	Payment Completion Status
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Payment Date

RELATIONSHIP SUMMARY

merchant (1) ⇨ product (M)
customer (1) ⇨ orders (M)
orders (1) ⇨ order_item (M)
order (1) ⇨ payment (1)