📘 Software Requirements Specification (SRS)

For: Order Management System

---

1. Introduction

1.1 Purpose

The purpose of this project is to develop an Order Management System (OMS) that helps businesses or customers manage product orders easily. The system will allow users to browse products, place orders, track order status, and manage order details efficiently.

1.2 Scope

The Order Management System automates the process of placing and managing customer orders.

It includes functionalities like:

- Viewing available products
- Adding products to the cart
- Placing and cancelling orders
- Viewing order history and details
- Admin managing products and orders

This system can be used by small shops or e-commerce startups to handle their order operations effectively.

1.3 Objectives

To simplify the order placement and tracking process.

- To reduce manual work in maintaining records.
- To provide real-time order status updates.
- To allow admin to manage products and track sales.

---

2. System Overview

The system will consist of two main modules:

1. Admin Module

- Manage product list (add, update, delete)
- View all customer orders
- Update order status

2. User Module

- View and search products
- Add to cart and place order
- Cancel orders and view history

---

3. Functional Requirements

ID	Requirement	Description
FR1	User Login/Signup	Users can create an account or log in.
FR2	Product Management	Admin can add, edit, or delete products.
FR3	View Products	Users can view all available products.
FR4	Add to Cart	Users can add items to their cart.
FR5	Place Order	Users can place an order for selected products.
FR6	Cancel Order	Users can cancel placed orders before dispatch.
FR7	Order History	Users can view previous orders.
FR8	Order Status	Admin can update and users can view order status (Pending, Shipped, Delivered, Cancelled).

---

4. Non-Functional Requirements

Category	Requirement
Performance	The system should handle multiple user requests simultaneously.
Usability	The interface should be user-friendly and simple to navigate.
Reliability	Data must be stored securely in the database.
Scalability	The system should allow future enhancements like payment gateway integration.
Security	Passwords must be encrypted; access control should be enforced for admin/user roles.

---

5. System Design

5.1 Modules

- Admin Panel: Product and Order Management
- User Panel: Product Viewing, Cart, Order, and History

5.2 Tools and Technologies

- Frontend: HTML, CSS, Bootstrap, JavaScript
- Backend: Java with Spring Boot
- Database: MySQL
- Server: Apache Tomcat
- IDE: IntelliJ IDEA / Eclipse

---

6. Database Design

Tables

1. product

Field	Type	Description
product_id	INT	Primary Key
name	VARCHAR	Product Name
price	DOUBLE	Product Price
quantity	INT	Available Quantity

2. order

Field	Type	Description
order_id	INT	Primary Key
user_id	INT	Foreign Key
order_date	DATE	Date of Order
status	VARCHAR	(Pending, Shipped, Delivered, Cancelled)

3. user

Field	Type	Description

user_id	INT	Primary Key
username	VARCHAR	Name of the user
password	VARCHAR	Encrypted password
email	VARCHAR	User email address

---

7. User Roles

Admin

- Login with admin credentials
- Manage product catalog
- View all orders and update their status

User

- Register and login
- Browse products and add to cart
- Place orders and view order status
- Cancel pending orders

---

8. Existing System

In the current (manual) system:

- Orders are taken and tracked manually.
- Records are maintained on paper or spreadsheets.
- Difficult to track order status and update inventory.

---

9. Proposed System

In the proposed automated system:

- All data is stored digitally.
- Users can place and track orders online.
- Admin can manage products and orders through a web interface.
- Reduces human error and saves time.

---

10. Conclusion

The Order Management System simplifies the process of placing, managing, and tracking orders. It is efficient, secure, and user-friendly, making it suitable for small or medium-scale businesses.
