**Software Requirements Specification (SRS)**

**Title Page**

**Project Title:** Watch E-commerce Website  
**Student Name(s) and Roll Number(s):** [Your Name(s), Roll Number(s)]  
**Course Name and Code:** [Your Course Name, Course Code]  
**Submission Date:** [DD/MM/YYYY]

**1. Introduction**

**1.1 Purpose**

The Watch E-commerce Website aims to provide an online platform where users can browse, search, and purchase watches. This document specifies the functional and non-functional requirements essential for the development and deployment of the website.

**1.2 Scope**

The platform will enable users to:

* View and filter a wide range of watches.
* Register, log in, and manage personal profiles.
* Add products to a shopping cart and place orders securely.
* Track orders and access customer support.
* Admin functionalities include managing products, tracking orders, and viewing sales reports.

**1.3 Stack Description**

* **Frontend:** HTML, CSS, JavaScript, React.js
* **Backend:** Node.js, Express.js
* **Database:** MongoDB
* **Tools:** Postman (API Testing), Git (Version Control)

**2. Overall Description**

**2.1 System Functions**

The system provides the following functions:

* User authentication and authorization.
* Product catalog with detailed descriptions and search/filter capabilities.
* Shopping cart for managing selected items.
* Payment processing and order confirmation.
* Admin panel for inventory and sales management.

**2.2 User Characteristics**

The platform is designed for:

* **End Users:** Individuals looking to purchase watches online.
* **Admins:** Responsible for managing the catalog and tracking orders.

**2.3 Assumptions and Dependencies**

* Users should have internet access and a compatible browser.
* Payment gateway integration requires third-party services like PayPal or Stripe.

**3. Functional Requirements**

**3.1 Core Features**

* **User Registration/Login:** Users can create accounts, log in, and manage profiles.
* **Product Browsing:** Browse watches with filtering options (e.g., brand, price).
* **Cart Management:** Add/remove products and update quantities.
* **Order Management:** Secure checkout and order tracking.
* **Admin Panel:** Manage products, view sales reports, and track orders.

**3.2 Software Requirements**

* **Frontend:** React.js, HTML, CSS, JavaScript
* **Backend:** Node.js, Express.js
* **Database:** MongoDB
* **Other Tools:** Postman, Git

**3.3 Hardware Requirements**

* **Processor:** Intel Core i5 or higher
* **RAM:** Minimum 8 GB
* **Storage:** Minimum 256 GB SSD

**4. Data Structures and Flow**

**4.1 Database Design**

| **Table Name** | **Fields** | **Description** |
| --- | --- | --- |
| Users | UserID, Name, Email, Password | Stores user information |
| Products | ProductID, Name, Price, Category | Stores product details |
| Orders | OrderID, UserID, ProductID, Date | Stores order information |
| Cart | CartID, UserID, ProductID | Manages items in user carts |

**4.2 Data Flow Diagram**

1. **User Interaction:** Users interact with the system to browse products or manage their carts.
2. **Backend Processing:** Server handles user requests and interacts with the database.
3. **Admin Interaction:** Admins manage inventory and orders via the admin panel.

**5. Non-Functional Requirements**

* **Performance:** The website should load within 2 seconds under normal conditions.
* **Scalability:** Should support up to 10,000 concurrent users.
* **Security:** User data and payment details must be encrypted.
* **Usability:** Provide a simple and intuitive interface.
* **Compatibility:** Compatible with major browsers and responsive on mobile devices.

**6. Appendices**

* **Acronyms:**
  + **API:** Application Programming Interface
  + **UI:** User Interface
* **References:**
  + PayPal Developer Documentation
  + Stripe API Documentation

**7. Submission Guidelines**

* File Format: PDF or DOCX
* Font Style: Times New Roman or Arial
* Font Size: 12 pt (body), 14 pt (headings)
* Line Spacing: 1.5
* File Name: SRS\_WatchEcommerce\_StudentName
* Submission Platform: Google Classroom
* Deadline: 02/01/2025