

**BCA261-2: Full Stack Development**

**Software Requirement Specification (SRS) Document**

by Prasanth (2441544)

Submitted to:

Prof/Dr. Rohini V

Assignment report submitted in partial fulfilment of the requirements for the Full Stack Development CIA 1(B) submission of CHRIST (Deemed to be University)

***Table of Contents***

|  |  |  |
| --- | --- | --- |
| ***S.no*** | ***Topic Name*** | ***PageNo*** |
| **1** | **Introduction** | **3** |
| **2** | **Overall Description** | **4** |
| **3** | **Functional Requirements** | **4** |
| **4** | **Data Flow Diagram** | **5** |
| **5** | **Non-Functional requirements** | **7** |
| **6** | **Appendices** | **7** |

**1. Introduction**

**1.1 Purpose**

The Watch E-commerce Website aims to provide a responsive website where users can browse, search, and purchase watches. This SRS provides a brief description about 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, MySQL
* **Tools:** Postman (API Testing), Git (Version Control)
* **Code Editor:** Visual studio code

**2.Overall Description**

**2.1 System Function:**

**The Main Functions of the website:**

* User authentication with authorization.
* Product catalog with description of the product like price.
* A cart page where the product added would be there.
* Payment processing and order conformation email.
* Admin panel for inventory and sales management.
* A separate webpage to show about the history and location of the company.

**2.2 Users Characteristics**

**The platform is designed for:**

* **End-Users:** For Consumers who are looking to purchase the products online.
* **Admin:** For checking the inventory and checking the sales and adding or removing products

**2.3 Assumptions and Dependencies**

**Following assumptions and dependencies for the website are:**

* Users need internet connection and proper browsers to access it.
* Payment gateway integration require third-party services from PayPal or Stripe Api

**3.Functional Requirements**

**3.1 Core Features**

* Product Browsing: Users can see various products by navigating through the Website.
* Cart: Whenever Users like the product and want to add it to cart there is a add to cart button when clicked the following products gets added to cart page.
* Sign-up/Sign-in: Users can make their own account and sign in through the website and make use of the perks for members.
* Payment gateway: There is secure payment for checking out and paying for the product

**3.2 Software Requirements**

* **Frontend:** Html, CSS, JavaScript, React.js
* **Backend:** Node.js, Express.js
* **Database:** MySQL, MongoDB
* **Other Tools:** GitHub, Postman

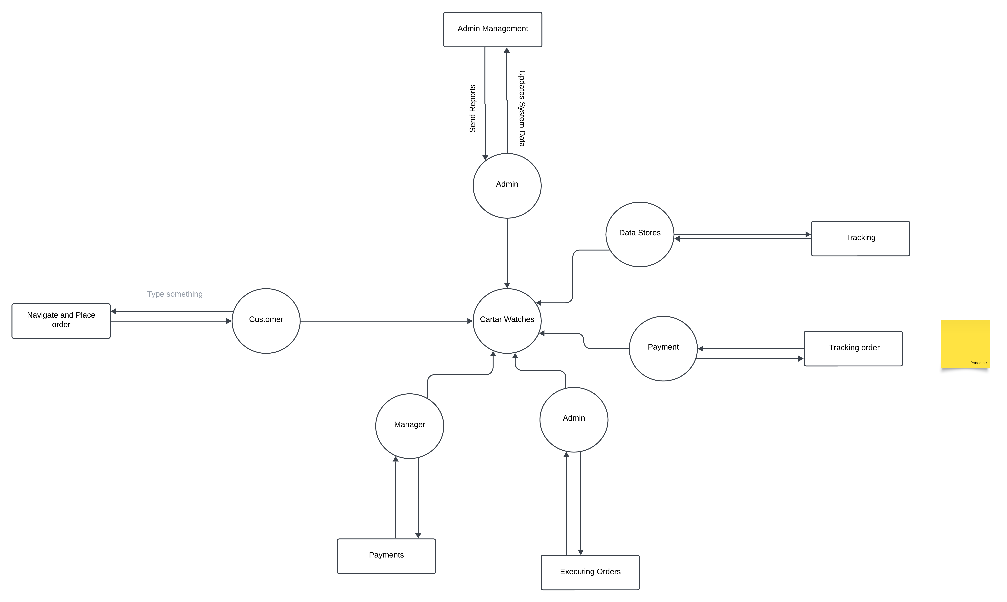
**3.3 Hardware Requirements**

* **Processor:** \*86 or arm processor
* **Storage:** Greater than 64 Gb
* **Ram:** More than 8 Gb

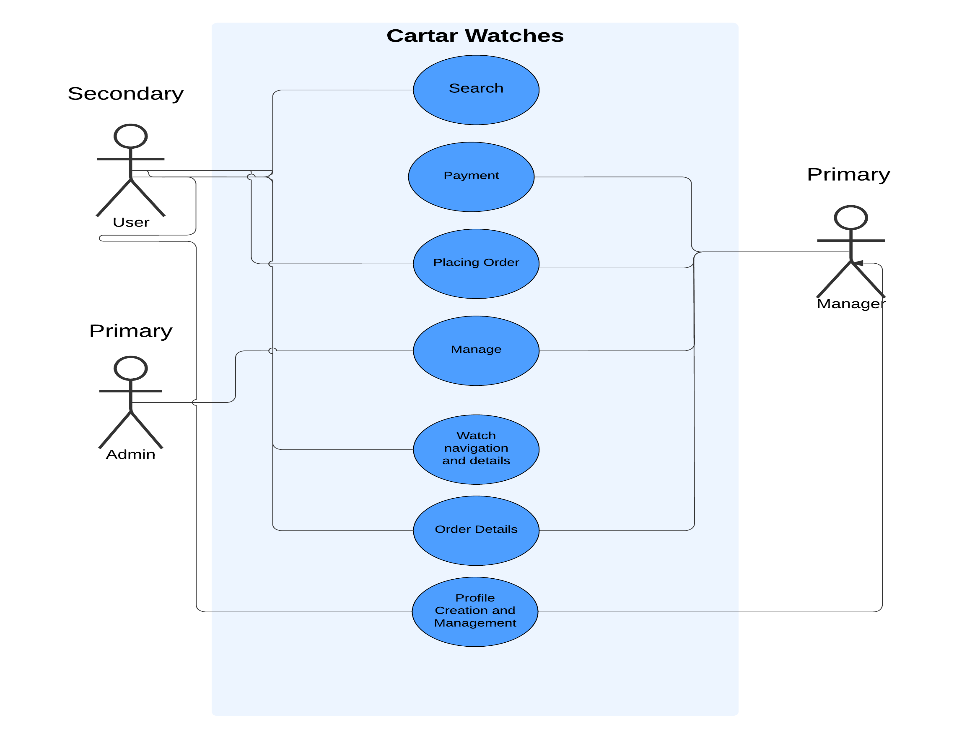
**4.Digrammatic Representation:  
Mind-map:**

****

**Data Flow Diagram:**

****

**Use Case Diagram:**

****

**5.Non-Functional Requirements**

* **Performance:** The website would able to load faster due to faster loading server.
* **Scalability:** As the backend would be stronger it would be able to host up to 10000 concurrent users
* **Security:** As a secure gateway payment is used there is less to no risk for customer data leak
* **Usability:** Provides a simple and user-friendly interface.
* **Compatibility:** It is compatible with every browser’s and even across devices

**6.Appendices**

* **Acronyms:**

**API:** Application programming interface

**UI:** User interface

* **References:**

PayPal Api documentation

Stripe Api documentation