**Software Requirement Specification**

**Project Name: Online Auction**

**1]Introduction**

* **Purpose**: The purpose of this SRS is to outline the software requirements for an online auction platform that facilitates the buying and selling of goods via an auction format.
* **Scope**: This platform will allow users to register, list items, bid on auctions, make payments, leave feedback, and manage shipping, all within a secure environment. Admin users will have tools for managing users, auctions, and site operations.

**2]** **System Overview**

The platform will consist of three user types: regular users (buyers/sellers), administrators, and support staff. Each user type will have different access and permissions.

**3] Functional Requirements**

**3.1** **User Registration and Authentication**

* **User Registration**: Users must be able to sign up with an email account.
* **User Login/Logout**: Provide authentication with password reset capabilities.
* **Profile Management**: Users can update their profiles, including contact details, payment information, and shipping addresses.

**3.2 Item Listings**

* **Create Listings**: Sellers can list items for auction, specifying the starting price, auction duration, and shipping details.
* **Listing Categories**: Items should be categorized for easier browsing.
* **Images and Descriptions**: Sellers can upload images and provide detailed descriptions.

**3.3** **Bidding System**

* **Real-time Bidding**: Users can place bids on items. The system will display real-time updates.
* **Bid Incrementing**: Allow automatic bid increments based on set thresholds.
* **Bid Notifications**: Notify users via email or in-app messaging when they have been outbid or won an auction.

**3.4** **Auction Management**

* **Start/End Auctions**: Admin can control auction start and end times.
* **Auction Status**: Display auction status (e.g., ongoing, completed, cancelled) for users.
* **Automatic Closing**: Auctions close automatically at the specified time, with the highest bidder winning.

**3.5** **Payment Processing**

* **Secure Payment Gateway**: Integration with secure payment gateways (e.g., PayPal, Stripe) for payment transactions.
* **Multiple Payment Methods**: Support for credit/debit cards, online wallets, and bank transfers.
* **Refunds**: Manage refunds in cases of cancelled transactions or returned items.

**3.6 User Feedback System**

* **Seller/Buyer Ratings**: Users can leave feedback for both buyers and sellers after each transaction.
* **Review Moderation**: Admins can moderate and remove inappropriate feedback.

**3.7** **Search and Filtering Options**

* **Search Bar**:Users can search for items by keywords, categories, and seller names.
* **Filter Options**: Filters for price range, auction end date, and item condition (new/used)

**3.8** **Messaging System**

* **Buyer-Seller Communication**: Enable messaging between buyers and sellers regarding items and auctions.
* **Admin Communication**: Provide users with a way to contact support or administrators.

**3.9** **Shipping Integration**

* **Shipping Options**: Sellers can specify available shipping methods and costs.
* **Shipping Tracking**: Integration with shipping services (e.g., FedEx, UPS) for tracking shipments.
* **Address Management**: Users can manage and update their shipping addresses.

**3.10** **Admin Dashboard**

* **User Management**: Admins can manage user accounts (ban, suspend, or delete users).
* **Auction Management**: Admins can create, edit, or remove auctions.
* **Payment and Transaction Management**: View and manage payment transactions and resolve disputes.

**3.11** **Legal Compliance**

* **User Agreement and Policies**: The platform should include a terms and conditions page, privacy policy, and refund/cancellation policies.
* **Data Privacy**: Ensure compliance with regulations like GDPR or CCPA for user data privacy.
* **Fraud Detection**: Implement tools to detect and prevent fraud.

**3.12** **Marketing Strategies**

* **Promotions and Discounts**: Allow sellers to offer discounts and promotional codes.
* **Email Marketing**: Integration with email services for promotional campaigns and updates.

**4. Non-Functional Requirements**

**4.1 Performance**

* The platform should handle high traffic during peak auction periods.
* The system must scale dynamically to accommodate user growth.

**4.2 Security**

* **Data Encryption**: All sensitive user data (passwords, payment details) should be encrypted.
* **Secure Transactions**: Ensure that all transactions are conducted over secure, encrypted channels.
* **User Authentication**: Implement two-factor authentication (2FA) for enhanced security.

**4.3 Usability**

* The platform should be easy to navigate, especially for new users.
* Provide tutorials or help sections for first-time users.

**5 Technologies to be used**

**Front-end**

* 1. **Technologies to be used:**

**Web Development:**

* + - HTML5:
      * HTML5 is the latest specification of the HTML language, and represented a major break with previous markup practices. The purpose of the profound changes to the language was to standardize the many new ways in which developers were using it, as well as to encourage a single set of best practices with regards to web development.
* CSS3

CSS, also known as Cascading Style Sheets, is a game-changing language that revolutionized the web design process. With its simple design, CSS makes it easy to enhance the presentation of web pages, by applying styles independently of the HTML that makes up each page. Essentially, CSS is responsible for the aesthetic appeal of a website, dictating everything from font choice to color scheme and spacing. With this language, developers and designers can create visually appealing websites that truly stand out.

* + - * React is a framework that employs Webpack to automatically compile React, JSX, and ES6 code while handling CSS file prefixes. React is a [JavaScript](https://www.simplilearn.com/tutorials/javascript-tutorial/introduction-to-javascript)-based UI development library. Although React is a library rather than a language, it is widely used in web development. The library first appeared in May 2013 and is now one of the most commonly used frontend libraries for web development.
      * React offers various extensions for entire application architectural support, such as Flux and React Native, beyond mere UI.

## Bootstrap5:

* + - * Bootstrap 5 is evolving with each release to better utilize CSS variables for global theme styles, individual components, and even utilities. We provide dozens of variables for colors, font styles, and more at a :root level for use anywhere. On components and utilities, CSS variables are scoped to the relevant class and can easily be modified.

## JavaScript :

* + - * [JavaScript](https://developer.mozilla.org/en-US/docs/Glossary/JavaScript) is a scripting language that enables you to create dynamically updating content, control multimedia, animate images, and pretty much everything else. (Okay, not everything, but it is amazing what you can achieve with a few lines of JavaScript code.)

## **Back-End:**

* + - **Python (Version 3.12.2):**
      * Python is a general-purpose, dynamically typed, high-level, compiled and interpreted, garbage-collected, and purely object-oriented programming language that supports procedural, object-oriented, and functional programming.
      * Because of its straightforward language framework, Python is easier to understand and write code in. This makes it a fantastic programming language for novices. Additionally, it assists seasoned programmers in writing clear and error-free code.
      * Python has many third-party libraries that can be used to make its functionality easier. These libraries cover many domains, for example, web development, scientific computing, data analysis, and more.

## Django Rest Framework (DRF)

* + - * Django REST framework is a powerful and flexible toolkit for building Web APIs.
      * The Web browsable API is a huge usability win for your developers.
      * [Authentication policies](https://www.django-rest-framework.org/api-guide/authentication/) including packages for [OAuth1a](https://www.django-rest-framework.org/api-guide/authentication/#django-rest-framework-oauth) and [OAuth](https://www.django-rest-framework.org/api-guide/authentication/#django-oauth-toolkit)2.
      * [Serialization](https://www.django-rest-framework.org/api-guide/serializers/) that supports both [ORM](https://www.django-rest-framework.org/api-guide/serializers#modelserializer) and [non-ORM](https://www.django-rest-framework.org/api-guide/serializers#serializers) data sources.
      * Customizable all the way down - just use [regular function-based](https://www.django-rest-framework.org/api-guide/views#function-based-views) [views](https://www.django-rest-framework.org/api-guide/views#function-based-views) if you don't need the [more](https://www.django-rest-framework.org/api-guide/generic-views/) [powerful](https://www.django-rest-framework.org/api-guide/viewsets/) [features](https://www.django-rest-framework.org/api-guide/routers/).
      * Extensive documentation, and great community support.
      * Used and trusted by internationally recognised companies including [Mozilla](https://www.mozilla.org/en-US/about/), [Red Hat](https://www.redhat.com/), [Heroku](https://www.heroku.com/), and [Eventbrite](https://www.eventbrite.co.uk/about/).

## **Database Management:**

Use database MySQL for data storage.

**6.Conclusion:**

This SRS document defines the key functional and non-functional requirements for developing an online auction platform that is secure, scalable, and user-friendly. The system will offer a comprehensive auction experience, from listing items to processing payments and managing shipping.