Online Auction System with Bidding History

Software Requirements Specification

INT219
Front End Web Developer

Names:

Shreya Tripathi

Registration No.:

12308254

Prepared for
Continuous Assessment 3
Spring 2025

Table of Contents

1. Introduction

This Software Requirements Specification (SRS) document provides a complete description of the software system being developed — the **Online Auction System**. It outlines the functional and non-functional requirements of the system, intended to serve as a guide for software engineers involved in designing, developing, testing, and maintaining the software.

This document aims to clearly define the software product's behavior, features, and constraints to ensure that all stakeholders have a shared understanding of the system. It addresses the needs of end users (buyers, sellers, and administrators), while also providing technical details required for implementation. The requirements described herein are intended to eliminate ambiguity and minimize the risk of system failure or misinterpretation during the development lifecycle.

The SRS is structured according to the IEEE standard, and contains detailed information on system features, interface descriptions, system models, use cases, user characteristics, performance requirements, and other relevant aspects. This document acts as a contract between stakeholders and developers, ensuring that the delivered product meets user expectations and business goals.

1.1Purpose

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed and comprehensive overview of the **Online Auction System**. This document defines the functional and non-functional requirements of the system, outlining how the platform enables users to create auctions, participate in bidding, and track bidding history. It is intended to serve as a reference for developers, testers, project stakeholders, and system administrators during the design, development, testing, and deployment phases.

The system aims to provide a user-friendly, secure, and efficient digital marketplace where buyers and sellers can interact through online auctions, ensuring transparency, real-time bid tracking, and proper administration of auction activities.

1.2 Scope

The **Online Auction System** is a web-based platform that facilitates the auctioning of items by registered users and allows others to place competitive bids on those items. It includes modules for:

- User registration and login
- Creation of auction listings
- Real-time bidding functionality
- Display and tracking of bidding history
- User profile management
- Administrative control over users and listings
- Email notifications for key events (e.g., bid success, auction expiry)

The system supports both buyer and seller roles within a unified user account. The admin panel allows for the management of users, monitoring of auctions, and maintaining overall system integrity. The application is built using PHP, MySQL, HTML, CSS, and JavaScript, and is designed to run on any standard web browser via a local or remote server.

1.3 Definitions, Acronyms, and Abbreviations

Auction	*A public sale where goods or services are sold to the highest bidder.*
Bid	*An offer made by a user to purchase an auctioned item at a specific price.*
User	*A registered participant in the system, acting as a bidder and/or seller.*
Admin	*A user with elevated privileges to manage the system and moderate activity.*
UI (User Interface)	*The graphical layout and interaction elements through which users access features.*
Database	*The structured storage of data used for managing users, auctions, and bids.*
PHP	*A server-side scripting language used to build the backend of the system.*
MySQL	*A relational database management system used for data storage and queries.*
SMTP	*Simple Mail Transfer Protocol used for sending emails from the system.*
Localhost	*A local server environment (like XAMPP) used to test the application offline.*

1.4 References

The following documents and resources have been referenced in the preparation of this Software Requirements Specification (SRS):

1. **IEEE Std 830-1998**, *IEEE Recommended Practice for Software Requirements Specifications*, IEEE, 1998.
2. **PHP Manual**, The PHP Group, available at: https://www.php.net/manual/en/
3. **MySQL Documentation**, Oracle Corporation, available at: https://dev.mysql.com/doc/
4. **MDN Web Docs**, Mozilla Foundation, available at: https://developer.mozilla.org
5. **XAMPP Guide**, Apache Friends, available at: https://www.apachefriends.org

1.5 Overview

This Software Requirements Specification (SRS) document is organized to provide a comprehensive description of the **Online Auction System**, covering both functional and non-functional aspects. The remaining sections of the document are structured to guide software developers, testers, and stakeholders through the technical requirements necessary to design, implement, and maintain the system.

2. General Description

This section provides a high-level overview of the Online Auction System, describing its context, functionality, and the environment in which it will operate. The purpose is to give readers a general understanding of the system without detailing specific technical requirements.

2.1 Product Perspective

The Online Auction System is a **standalone web-based application** that functions independently, though it may be integrated with email services (SMTP) for notifications. It is developed using commonly available web technologies — **PHP**, **MySQL**, **HTML**, **CSS**, and **JavaScript** — and runs on local servers (like XAMPP) or remote hosting environments that support Apache, PHP, and MySQL.

This system is not a module of any larger software product, but it follows the architectural patterns of typical e-commerce and auction platforms. It interacts with users through a browser-based user interface and stores all data in a structured relational database.

2.2 Product Functions

The system performs the following key functions:

- Allows users to **register**, **log in**, and **manage profiles**.
- Enables users to **create, view, and manage auction listings**.
- Facilitates **bidding** on active auctions and dynamically updates bid status.
- Maintains a **bidding history** for transparency and user review.
- Sends **email notifications** for bid confirmations, auction results, and alerts.
- Provides an **admin panel** to manage users, auctions, and monitor activity.
- Ensures data is stored securely and presented in a user-friendly manner.

2.3 User Characteristics

There are two main types of users in the system:

- **General Users** (Buyers/Sellers):
    - May have basic to moderate technical proficiency.
    - Should be able to navigate the web-based UI with ease.
    - Will use the system to bid, list items, and track auctions.
- **Administrators**:
    - Have higher access privileges to moderate content and manage user activity.
    - Should understand basic platform management functions.
    - Are responsible for maintaining system integrity and enforcing rules.

No specialized training is required to use the system, but users are expected to have access to a stable internet connection and a modern web browser.

2.4 General Constraints

- The system must be developed using **PHP (v7.4 or higher)** and **MySQL**.
- The frontend must be compatible with modern browsers (Chrome, Firefox, Edge).
- The backend must run on **Apache server** (tested using XAMPP).
- The system should operate efficiently on **localhost** or shared hosting environments.
- Email features depend on proper SMTP configuration and internet access.
- All components must use **open-source technologies**.

2.5 Assumptions and Dependencies

- *It is assumed that the deployment environment supports **PHP and MySQL**.*
- *It is assumed that **SMTP service** (e.g., Gmail or other SMTP server) is available for email notifications.*
- *The system depends on the **Apache server** for execution; performance may vary based on server configuration.*
- *Users will access the system through a **stable internet connection** using **modern web browsers**.*
- *It is assumed that users will have access to **basic input devices** (mouse, keyboard) and **standard resolution displays**.*

3. Specific Requirements

This section details all the technical and functional specifications that must be met by the Online Auction System. The information provided will guide software development, testing, and validation, ensuring that the system behaves as expected under defined constraints and conditions.

3.1 External Interface Requirements

3.1.1 User Interfaces

The system shall offer a responsive and intuitive user interface accessible via standard web browsers. It will consist of:

- **Home Page**: Displays featured or ending-soon auctions.
- **User Authentication Interface**: Login and registration forms with validation prompts.
- **Dashboard**:
  - **Buyers**: View current bids, auction history, and notifications.
  - **Sellers**: Manage created auctions, monitor bids.
- **Auction Listing Page**: Form to create new auctions including fields like item name, description, starting price, deadline, and item image upload.
- **Bidding Interface**: Allows users to view item details and place bids. Real-time updates reflect current highest bid.
- **Admin Panel**: Role-specific dashboard for admin users to manage users, review flagged content, and oversee auction integrity.
- **Error Pages**: Custom 404 and system error pages to guide users.

All interfaces should prioritize **usability, responsiveness**, and **accessibility** across different screen sizes.

3.1.2 Hardware Interfaces

- ➢ **Server Requirements**:
- The system should be hosted on a web server with at least the following configuration:
  - Processor: 2.0 GHz or higher, multi-core processor.
  - RAM: Minimum 4GB of RAM for smooth operations.
  - Storage: At least 10GB of free space for storing database files, user data, and auction images.

    - Network: Reliable internet connection with minimum upload/download speed of 10 Mbps.
- ➢ **Client Devices**:
- The system should be accessible on modern desktop and mobile devices. The supported browsers include Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge.
- Mobile accessibility is critical, and the system should be responsive to adapt to different screen sizes.

3.1.3 Software Interfaces

- **Database Interface**:
    - The system will interact with a MySQL database for storing all information related to users, auctions, bids, and system logs.
    - Tables to be included:
        - **Users**: Storing user details like name, email, password, etc.
        - **Auctions**: Storing details of auctions such as title, description, images, price, and auction end time.
        - **Bids**: Recording the bids placed by users for each auction.
        - **Transactions**: Recording auction completion and payments.
- **Email Service Interface**:
    - The system will use an SMTP service (such as Gmail, SendGrid, etc.) to send transactional emails, including:
        - **Password Reset Emails**: Sent when a user requests a password reset.
        - **Auction Notifications**: Notifying users about auction status (win/loss).
        - **Registration Confirmation**: Email confirmation upon successful registration.

3.1.4 Communications Interfaces

- **HTTP/HTTPS**:
    - Communication between the user and the server will be handled over HTTP and HTTPS protocols. HTTPS will be used for secure communication, especially when handling sensitive data such as login credentials and personal information.
- **Email Handling**:
    - The system will integrate with an SMTP server for sending and receiving emails. It should support both text-based and HTML emails for better user engagement.

3.2 Functional Requirements

The functional requirements define the specific behavior and capabilities of the system. These requirements describe the features of the Online Auction System, ensuring that it meets the needs of both buyers and sellers. Each feature is broken down into specific inputs, processing steps, outputs, and error handling mechanisms, ensuring a seamless and functional user experience.

3.2.1 User Authentication

3.2.1.1 Introduction

User authentication ensures that only authorized users can access the system. This includes login, logout, and password recovery functionalities. Proper authentication will help maintain the security and privacy of users' accounts and bidding activities.

3.2.1.2 Requirement

1. **Login Functionality:**
   - Description: The system must allow users to log into their accounts using a registered email address and password.
   - Inputs: Email/username and password.
   - Processing: The system must authenticate the credentials by matching them with the information stored in the database.
   - Outputs: Upon successful login, the user is redirected to the homepage or profile page. An error message should appear if login fails.
   - Error Handling: Display an error message such as "Invalid username/password."
2. **Logout Functionality:**
   - Description: Users should be able to log out from their session.
   - Inputs: A logout button.
   - Processing: The system must invalidate the user's session upon logout.
   - Outputs: The user will be redirected to the homepage, and all session data will be cleared.
   - Error Handling: If logout fails, an error message should appear, and the user should be redirected back to the homepage.
3. **Password Recovery:**
   - Description: Users must be able to recover their password if forgotten.
   - Inputs: Email address.
   - Processing: The system sends a password reset link to the registered email address.
   - Outputs: A password reset email with a secure link for the user to reset their password.
   - Error Handling: If the email is not registered, display a message "Email not found in the system."

3.2.2 Auction Management

3.2.2.1 Introduction

Auction management is a core functionality of the system that allows sellers to create auctions and buyers to participate by placing bids. This section ensures that auctions are listed correctly and that users can create and manage their auctions efficiently.

3.2.2.2 Requirement

1. **Create Auction:**
   - Description: Sellers must be able to create auctions with specific details like the title, description, starting price, auction duration, and images.
   - Inputs: Title, description, category, price, images, and auction end date.
   - Processing: The system validates the inputs (e.g., ensuring the price is a positive number and the end date is in the future), then stores the auction in the database.
   - Outputs: A confirmation message that the auction has been successfully created.
   - Error Handling: Display an error message if any input is invalid, such as "Starting price must be greater than 0" or "Auction end time must be in the future."
2. **View Active Auctions:**
   - Description: Users should be able to view all active auctions and filter them based on categories like price, duration, and product type.
   - Inputs: Filters (e.g., price range, category).
   - Processing: The system retrieves auctions that match the user's selected filters from the database and displays them.
   - Outputs: A list of active auctions, sorted or filtered according to user preferences.
   - Error Handling: If no auctions match the filters, display a message like "No auctions found for the selected filters."
3. **Bid on Auctions:**
   - Description: Registered users must be able to place bids on active auctions.
   - Inputs: Bid amount.
   - Processing: The system ensures the bid is higher than the current highest bid. If valid, the bid is stored in the database and updates the auction’s current bid.
   - Outputs: A success message with the updated highest bid.
   - Error Handling: If the bid amount is too low (i.e., less than the current bid), display an error: "Bid must be higher than the current bid."

3.2.3 Bidding History

3.2.3.1 Introduction

The bidding history feature allows users to track the progress of their bids, ensuring transparency and accountability in the auction process. Both buyers and sellers need access to detailed bidding histories.

3.2.3.2 Requirement

1. **Display Bid History:**
   - Description: Users should be able to view a history of all their bids, including the auction title, bid amount, time placed, and whether they won or lost the auction.
   - Inputs: None (automatically generated based on user activity).
   - Processing: The system retrieves the user’s bid history from the database and displays it in a readable format.
   - Outputs: A list of bids placed by the user, showing the auction, bid amount, date/time, and outcome (win/loss).
   - Error Handling: If there are no bids placed, display a message: "You have no bidding history."
2. **View Auction Status:**
   - Description: Users should be able to view the current status of an auction they have participated in (e.g., the current highest bid, time remaining).
   - Inputs: Auction ID (selected by user).
   - Processing: The system fetches the latest bid information for the selected auction.
   - Outputs: The current highest bid, remaining time, and number of bids placed.
   - Error Handling: If the auction is not found, display an error: "Auction not found."

3.5 Non-Functional Requirements

Non-functional requirements define the overall attributes of the system, including its performance, reliability, availability, security, maintainability, and portability. These requirements focus on how well the system operates and the constraints under which it must function. While functional requirements describe the system's behavior and features, non-functional requirements ensure that the system performs efficiently, securely, and is adaptable for future growth.

3.5.1 Performance

Performance requirements define how quickly and efficiently the system should respond to user requests and handle a high volume of activity.

- **Response Time**: 95% of all user requests should be processed in under 2 seconds. Pages such as auction listings, bidding history, and user profiles should load within 3 seconds to ensure a seamless user experience.
- **Throughput**: The system should support a minimum of 100 active users concurrently without noticeable performance degradation. This includes activities like browsing auctions, placing bids, and accessing user profiles.

3.5.2 Reliability

Reliability refers to the system's ability to perform its required functions consistently over time, with minimal downtime.

- **System Availability**: The system must be available 99.5% of the time, allowing for up to 5 hours of downtime per month for scheduled maintenance or unforeseen issues.
- **Error Handling**: The system must be able to gracefully handle errors without crashing, ensuring that users are informed of issues (e.g., a maintenance page or error message).

3.5.3 Availability

Availability ensures that the system is accessible and operational when needed.

- **24/7 Availability**: The system must be available 24 hours a day, 7 days a week, except for scheduled maintenance windows.
- **Scheduled Maintenance**: Maintenance periods must be communicated to users in advance via notifications on the website and emails. Downtime for maintenance should not exceed a predefined limit, and emergency outages should be minimized.

3.5.4 Security

Security requirements outline how the system will protect user data and ensure privacy, preventing unauthorized access or data breaches.

- **Encryption**: All sensitive user data (including passwords, personal details, and bidding history) must be encrypted using AES-256 or a stronger encryption algorithm to ensure confidentiality.
- **Data Privacy**: The system must handle user data in compliance with data protection regulations, including GDPR (General Data Protection Regulation). User data should be stored securely, and user consent must be obtained before collecting any personal information.

3.5.5 Maintainability

Maintainability ensures that the system is easy to update, debug, and extend with new features as needed.

- **Modular Codebase**: The system's codebase should be modular, with well-defined components that can be easily updated or replaced without affecting other parts of the system. Each module should have clear documentation and appropriate comments.
- **Error Logging**: The system should incorporate logging mechanisms to monitor and log system errors, user activities, and overall health. This will help developers identify issues and ensure smooth functioning.
- **Automated Testing**: Unit and integration tests should be developed to automatically check for common bugs and ensure that new code changes do not introduce regressions.

3.5.6 Portability

Portability ensures that the system can be deployed on various platforms without major adjustments.

- **Server Environment**: The system should be portable to various server environments that support PHP and MySQL. This means the system should be able to run on different operating systems (e.g., Linux, Windows) with minimal configuration changes, ensuring smooth deployment across different hosting providers or cloud services.

3.7 Design Constraints

- The system must be designed using PHP, MySQL, and JavaScript.
- It must be mobile-friendly, ensuring a responsive design on both desktop and mobile devices.
- The system must comply with industry standards for e-commerce platforms, such as secure payment processing and user authentication.

3.9 Other Requirements

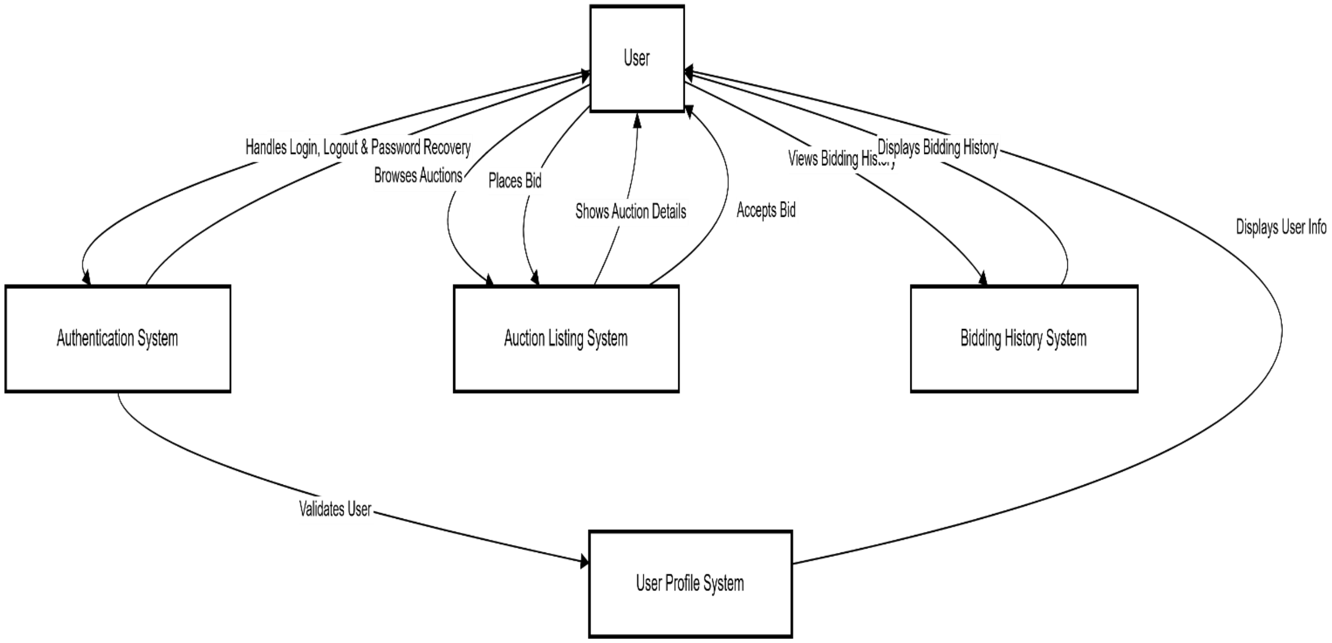
- **Language Support**: The system should support multiple languages, starting with English and the potential to add more languages in the future.
- **Legal Compliance**: The system must comply with local e-commerce regulations and consumer protection laws.

4. Analysis Models

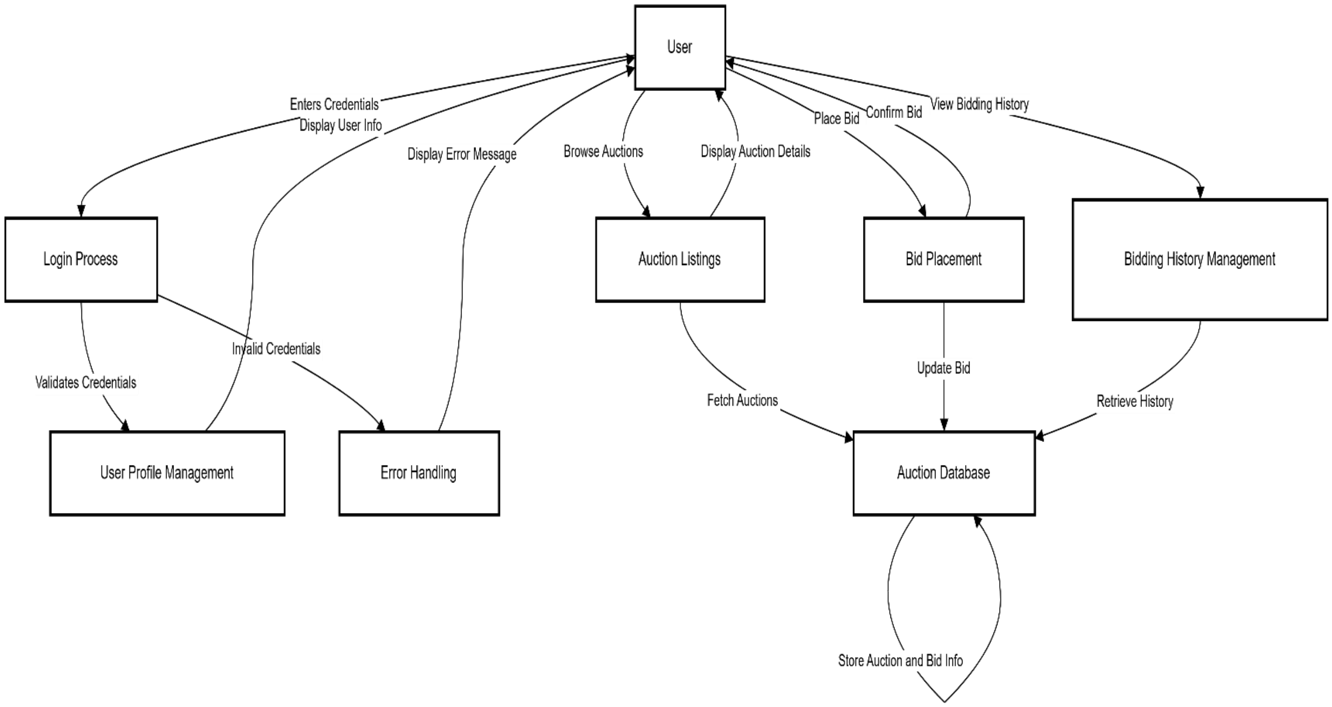
In this section, we describe the various analysis models used to define and refine the functional requirements of Bidpulse. Each model helps visualize the flow of data, the interaction of the system with external entities, and the internal structure of the software.

4.1 Data Flow Diagrams (DFD)

Level 0 DFD:



Level 1 DFD:



5. GitHub Link

https://github.com/Soumyosish/auction_system

A. Appendices

The following appendices provide supporting information relevant to the development and understanding of the Online Auction System. While these appendices offer additional insights and background, only those specifically referenced in the body of the SRS are to be considered part of the formal set of system requirements.

A.1 Appendix 1 – Initial Concept Document

This document outlines the original concept for the Online Auction System. It includes a high-level overview of user needs, initial feature ideas, and the core objectives of the system, such as:

- Enabling users to browse and participate in online auctions
- Providing real-time bidding updates
- Allowing users to view their bidding history
- Ensuring secure user authentication

This conceptual draft served as the foundation for the functional and non-functional requirements specified in this SRS.