Project Title

AuctionEase

Version 1.0

Course: ACS 56000 Software Engineering

Initial development focusing on core auction functionality, user account management, real-time bidding, real-time notifications, and secure transaction processing.

Table of Contents

Introduction

AuctionEase is an online auction platform designed to make real-time bidding and seamless transactions between buyers and sellers. The system allows users to create accounts, list items for auction, place bids in real-time, and receive instant notifications about their bidding activities. Sellers benefit from a streamlined process to manage their auction listings, set parameters like reserve prices and auction end times, and engage with potential buyers through integrated communication tools.

The platform leverages a robust MySQL database to securely store user information, auction listings, and bid histories. By incorporating features such as third-party authentication, real-time notifications, and comprehensive reporting tools, AuctionEase aims to provide a reliable and user-friendly environment for conducting online auctions. The system is built using Java Spring Boot for the server-side, ensuring scalability and performance, and is accessible via both web and mobile applications to cater to users' diverse needs.

Glossary

- **Auctioneer:** The user who lists items for auction.
- **Bidder:** A user who places bids on auctioned items.
- **Real-Time Bidding:** A live bidding process where bids are updated instantaneously.
- **Reserve Price:** The minimum price a seller is willing to accept for an auctioned item.
- **Buy It Now:** A feature allowing users to purchase an item immediately at a fixed price, bypassing the bidding process.
- **Watchlist:** A personalized list where users can monitor auctions without actively bidding.
- **ETL Pipeline:** A process that extracts data, transforms it, and loads it into a database.
- **CI/CD:** Continuous Integration and Continuous Deployment, practices that enable frequent and reliable software releases.
- **Role-Based Access Control (RBAC):** A security mechanism that restricts system access based on user roles.
- **Third-Party Authentication:** Allowing users to log in using external services like Google or Facebook.
- **Real-Time Notifications:** Instant alerts sent to users about important events related to their activities.
- **Unit Testing:** Testing individual components of the software to ensure they work correctly.
- **Integration Testing:** Testing combined parts of the application to evaluate their interaction.
- **End-to-End Testing:** Testing the complete workflow of the application from start to finish.
- **Kafka:** A distributed event streaming platform used for building real-time data pipelines.

User Requirements

1. **Account Creation:**
Users can create an account by providing necessary details such as email, password, and personal information.

2. **Item Listing for Auction:**
Users can list items for auction with detailed descriptions, multiple images, and starting bid prices.

3. **Browsing and Searching:**
Users can browse and search available items listed for auction using various filters and search criteria.

4. **Real-Time Bidding:**
Users can place bids on items of interest in real-time, with instant updates on bid status.

5. **Bidding History and Status:**
Users can view their bidding history and monitor the status of their current bids.

6. **Outbid Notifications:**
Users receive real-time notifications if they have been outbid on an item they are bidding on.

7. **Auction Win Notifications:**
Users are notified immediately if they win an auction.

8. **Detailed Auction History:**
Users can access detailed auction history, including final bid prices and information about the winning bidder.

9. **Auction Configuration by Sellers:**
Sellers can set auction end times and minimum reserve prices for their listed items.

10. **Password Recovery:**
Users can reset their account passwords through a secure recovery process.

11. **Countdown Timers:**
The system displays countdown timers for live auctions, showing time remaining until auction closure.

12. **Data Export:**
Users can export their bidding and auction data for external use, such as personal records or analysis.

13. **Image Upload for Sellers:**
Sellers can upload multiple high-quality images for their auction listings to enhance item visibility.

14. **Third-Party Authentication:**
Users can authenticate their accounts through third-party platforms such as Google or Facebook for convenience and security.

15. **Following Sellers and Categories:**
Users can follow specific sellers or categories to receive notifications about new auctions and updates.

16. **Auction Filtering:**
Users can filter auctions based on parameters like time remaining, item category, condition, seller rating, geographical location, or bid amount.

17. **Automatic Bidding:**
Users can schedule automatic bids up to a predefined maximum amount to stay competitive in auctions without constant monitoring.

18. **Watchlist Feature:**
Users can save items to a "Watchlist" to monitor auctions without committing to bidding immediately.

19. **Seller Reviews and Ratings:**
After a successful auction, users can provide reviews and ratings for sellers to build trust and community reputation.

20. **Bid Withdrawal:**
Users can withdraw their bids before the auction ends, subject to a penalty fee.

21. **Periodic Summary Emails:**
Users receive periodic summary emails detailing their auction activities, including bids placed and auctions participated in.

22. **Buy It Now Option for Sellers:**
Sellers can mark items as "Buy It Now," allowing users to bypass the bidding process and purchase items immediately at a fixed price.

23. **Detailed Activity Reports:**
Users can request detailed reports of their bidding and purchasing activities for purposes such as tax filing.

24. **Seller Communication:**
Users can chat with sellers to obtain additional details about listed items, ensuring informed bidding.

25. **Mobile Accessibility:**
Users can access the platform via a mobile application, enabling on-the-go bidding and monitoring.

26. **Notification Preferences Management:**
Users can manage their notification preferences, choosing between email, SMS, and in-app notifications.

27. **Participation History:**
Users can view their participation history, including auctions they watched but did not bid on.

28. **Private Auctions:**
Sellers can set up private auctions where only invited users can participate, ensuring exclusivity.

29. **Advanced Filtering Options:**
Users can filter items by condition (new, used, refurbished), seller rating, and geographical location to find items that meet their specific criteria.

System Requirements

1. **Real-Time Notification System:**
The system sends notifications for real-time events, such as outbids, auction wins, and new auction listings, ensuring users are promptly informed.

2. **Secure MySQL Database:**
A MySQL database securely stores all user details, auction listings, and bid histories, employing encryption and secure access protocols to protect sensitive information.

3. **Role-Based Access Control (RBAC):**
RBAC will be implemented to ensure that only authenticated users can participate in auctions or list items for sale, with distinct permissions for bidders and sellers.

4. **Third-Party Payment Integration:**
The system integrates with third-party payment gateways like PayPal and Stripe to facilitate secure and reliable financial transactions between users.

5. **Technology Stack:**
    - **Server:** Java Spring Boot
    - **Client:** Vaadin

6. **Deployment Environment:**
The server runs on Ubuntu, with the option to deploy on a Virtual Machine (VM) to ensure scalability and reliability.

7. **Automated Backup and Recovery:**
Implementing automated backup and recovery processes to prevent data loss, ensuring business continuity and data integrity.

8. **Comprehensive Testing:**
Incorporating unit testing, integration testing, and end-to-end testing to ensure system reliability and performance across all components.

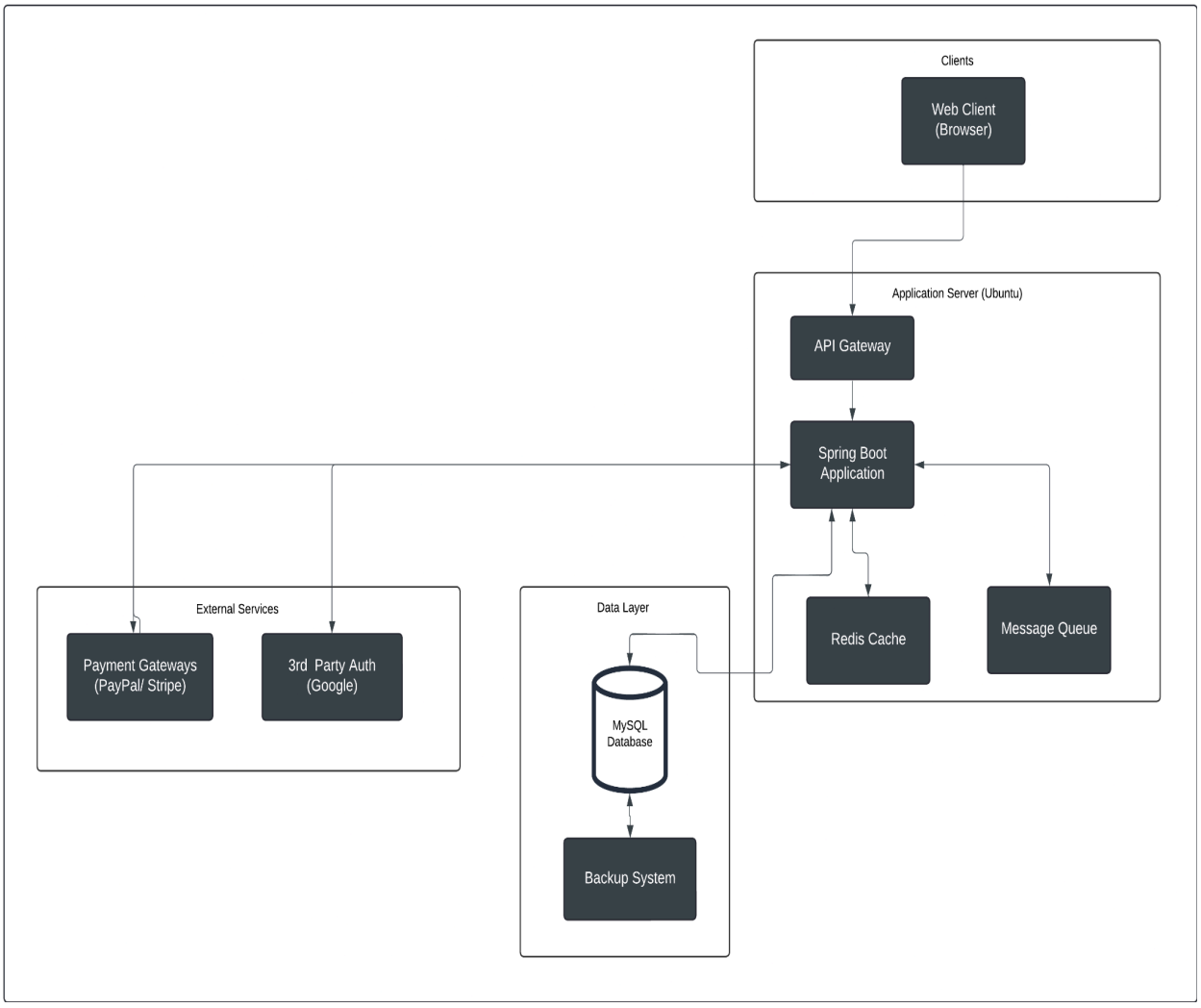
9. **CI/CD Pipelines:**
Utilizing Continuous Integration and Continuous Deployment pipelines to automate the deployment process and facilitate regular updates and maintenance.

10. **Data Handling Enhancements:**
Eventing System is used for publishing notifications and handling real-time data updates seamlessly.

5. System Architecture

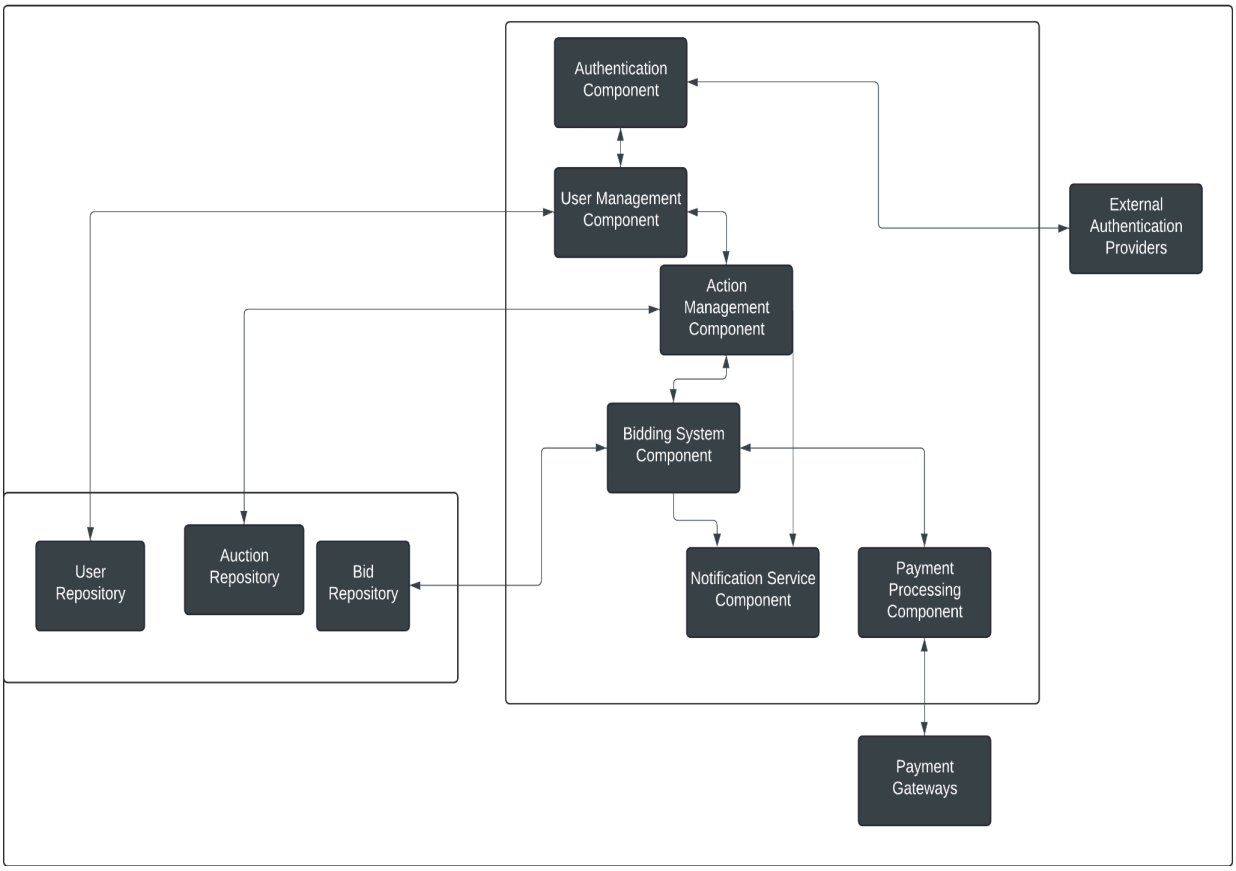
5.1 Architecture Diagram

Below is the high-level architecture diagram illustrating the main components of the AuctionEase system, including the client applications (web and mobile), server-side applications built with Java Spring Boot, the MySQL database, third-party payment gateways, and the real-time notification service is represented below.



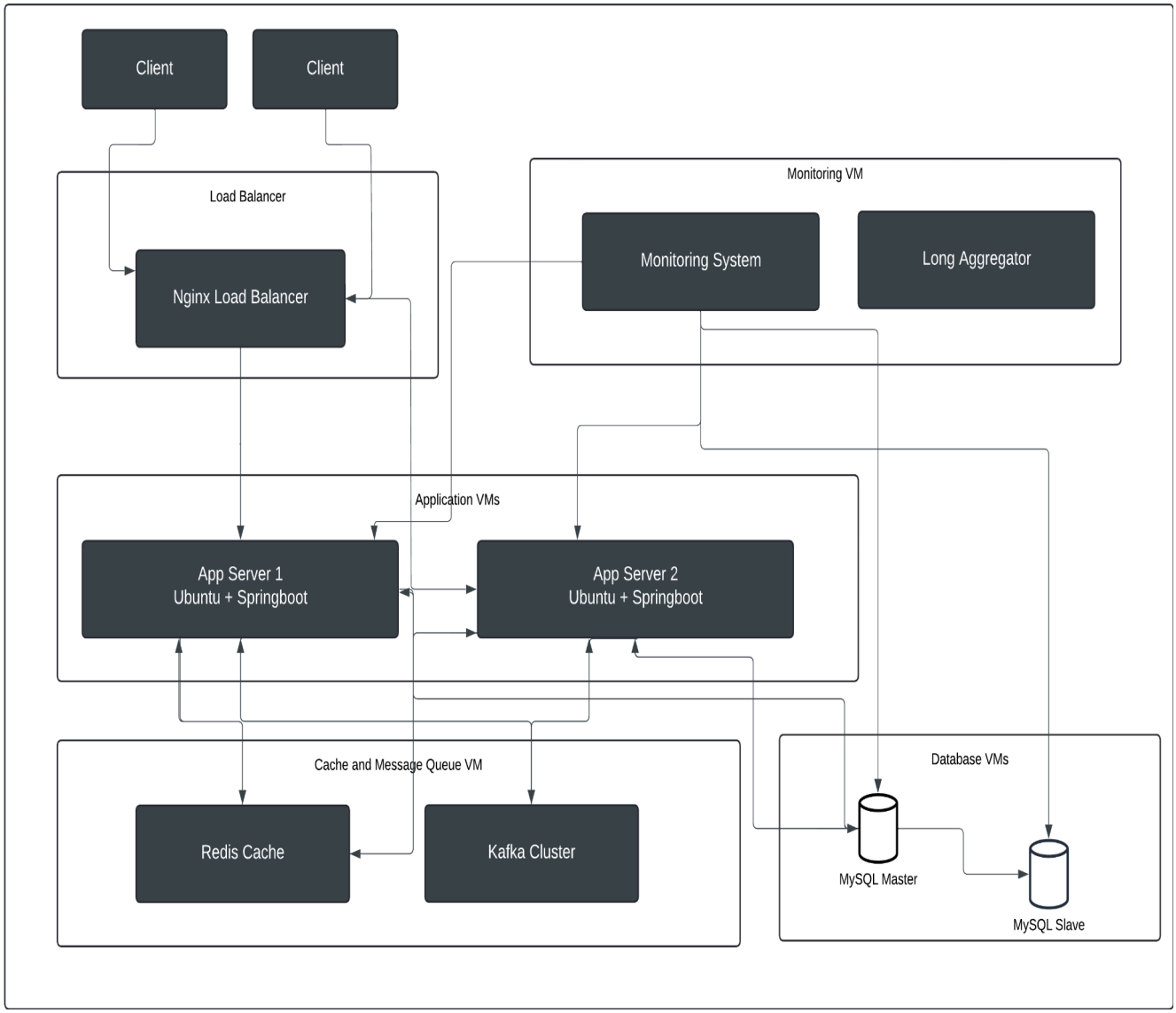
5.2 Component Diagram

A detailed component diagram includes the internal modules of the server application, such as user authentication, auction management, bidding system, notification service, and payment processing. It also includes how these components interact with each other and with external services like payment gateways and third-party authentication providers.



5.3 Deployment Diagram

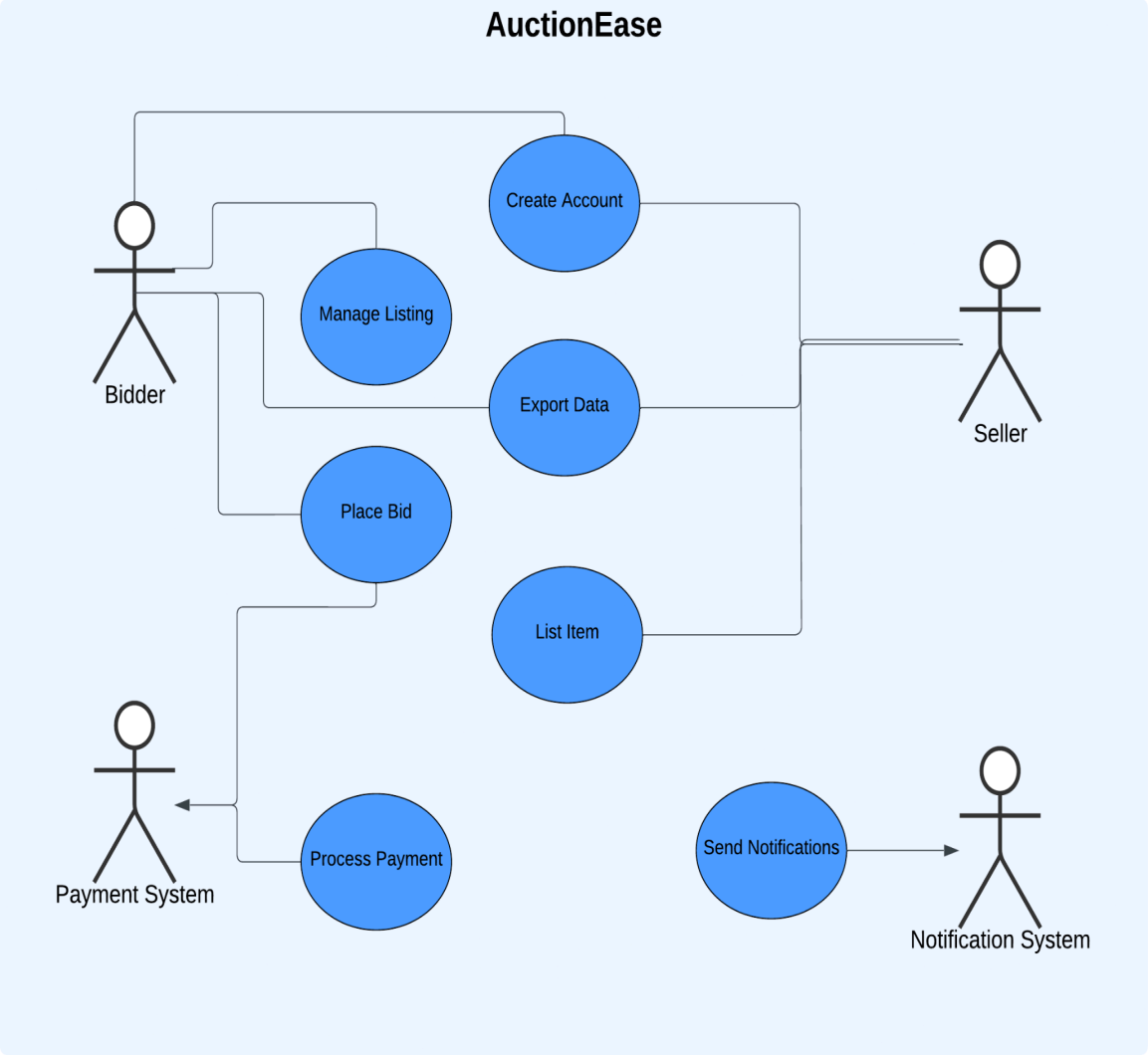
The deployment diagram shows the physical deployment of the AuctionEase system on Ubuntu servers, including the application server, database server, and any auxiliary services like backup systems and caching mechanisms. It also depicts the use of Virtual Machines.



System Models

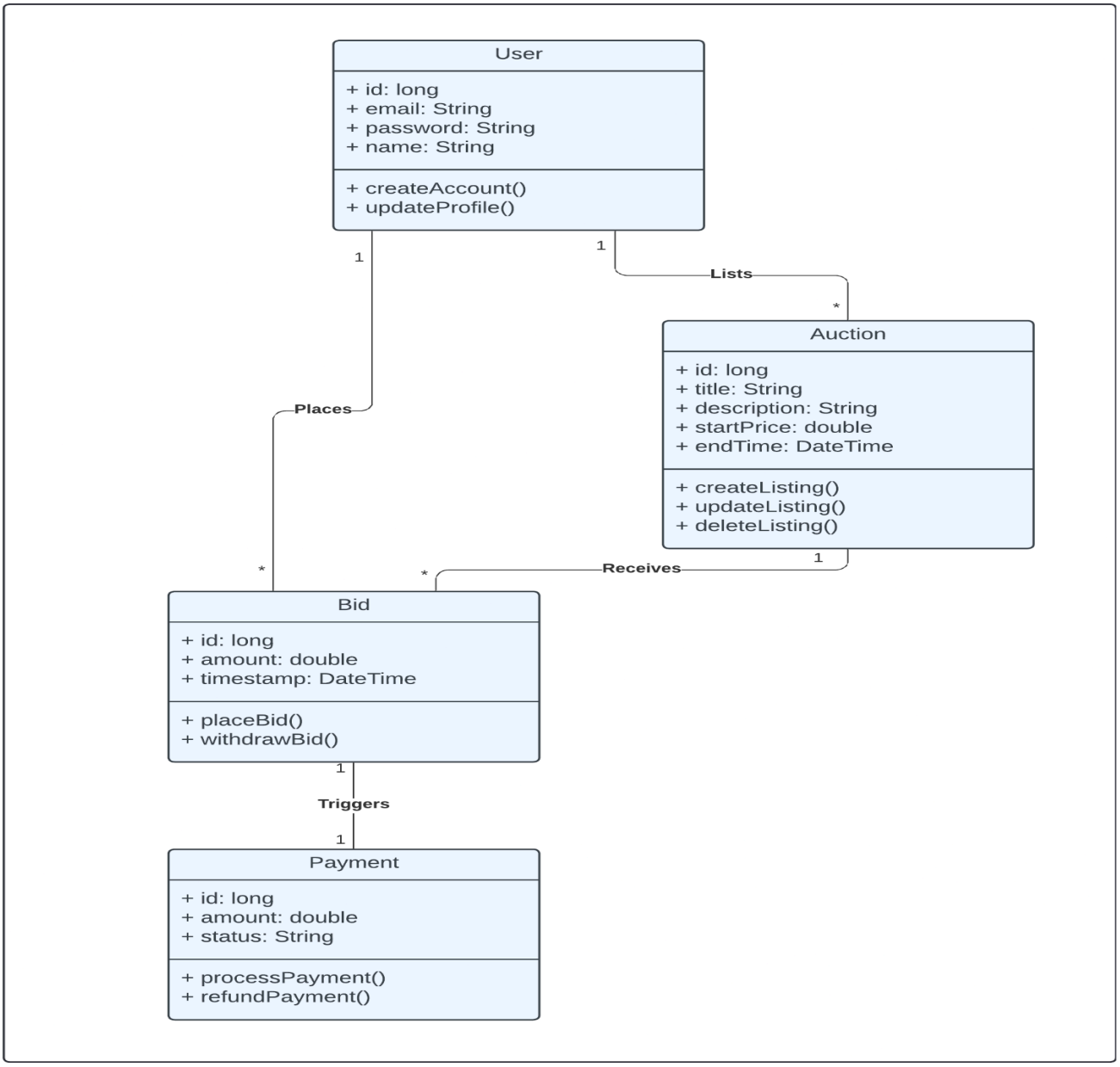
6.1 Use Case Diagram

This use case diagram includes the interactions between users (bidders and sellers) and the system. Key use cases include account creation, listing items, placing bids, receiving notifications, managing watchlists, and processing payments.



6.2 Class Diagram

The class diagram represents the data models and their relationships within the system. Important classes may include User, Auction, Bid, Payment, Notification, and Review. This diagram illustrates attributes and methods for each class, as well as associations, inheritances, and dependencies.



6.3 Sequence Diagram

This sequence diagram illustrates the process of placing a bid in real-time. It depicts the interaction between the user interface, server application, database, and notification service, showing the sequence of messages exchanged when a user places a bid, the bid is processed, and notifications are sent out.