Team S

Big Bidness
Software Requirements Specification
for Mini E-bidding System

**Simona Isakova, Ishmam Fardin, Brandon Vasquez,
Anas Ahmed, Nirath Hussan**

Version 1.0

Revision History

Date	Version	Description	Author
15/10/24	1.0.0	Added Purpose & Scope for Section 1.1 and 1.2 respectively	Ishmam
15/10/24	1.1.0	Added Definitions & Acronyms for Section 1.3	Ishmam
15/10/24	1.1.1	Added Actors and Use Cases for Section 2.1	Ishmam
15/10/24	1.1.2	Added Assumptions & Definitions for Section 2.2	Ishmam
15/10/24	1.1.3	Create Use Case Report Table & Added Supplementary Requirements for Sections 3.1 and 3.2 respectively	Ishmam
15/10/24	1.1.4	Created Overview for Section 1.5	Ishmam
15/10/24	1.1.5	Added additional Assumptions & Definitions for Section 2.2	Brandon
15/10/24	1.1.6	Restructured Use-case Reports table with IBM Use-case template	Brandon
15/10/24	1.1.7	Added IBM template resource to References subsection 1.4	Brandon
15/10/24	1.1.8	Created the Index to Supporting information	Anas
15/10/24	1.1.9	Added Appendices	Anas
16/10/24	1.2.0	Added onto Purpose & Scope for Section 1.1 and 1.2 respectively	Simona
16/10/24	1.21	Added additional Definitions & Acronyms for Section 1.3	Simona
08/11/24	2.0.0	Added petri net diagram	Brandon
08/11/24	2.0.0	Kick Started GitHub	Brandon
08/11/24	2.0.0	Created Boilerplate for Frontend and Backend	Ishmam
08/11/24	2.0.0	Updated Table of Contents, Text Formatting, and Meeting Notes	Nirath
10/11/24	2.0.0	Added ER diagram	Nirath
10/11/24	2.0.0	Added User case diagram	Simona
10/11/24	2.0.0	Created Navbar	Ishmam
11/11/24	2.0.0	Added Sequence class diagram	Ishmam
11/11/24	2.0.0	Added most function Pseudocode	Anas
11/11/24	2.0.0	Added Figma Images	Ishmam
11/11/24	2.0.0	Added Class Collaboration Diagram	Nirath
11/11/24	2.0.0	Added Flowchart Diagram	Brandon

Mini E-bidding System	Version: 2.0
Software Requirements Specification	Date: 12/11/24
Team S	

Table of Contents

Mini E-bidding System	Version: 2.0
Software Requirements Specification	Date: 12/11/24
Team S	

Software Requirements Specification

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the external behavior and functionalities of an e-bidding platform that facilitates seamless online transactions. This document outlines the system's key user roles, including Visitors, Users (VIP and non-VIP), and Super-users, along with their respective capabilities and responsibilities. The platform enables users to browse, list, bid on, and transact goods or services, all while maintaining secure and efficient payment processing. A feedback system, user moderation mechanisms, and personalized user interfaces are provided to ensure a reliable, user-friendly experience. Additionally, it outlines constraints related to user suspension and reinstatement rules to foster a fair and balanced transaction environment. Ultimately, this specification aims to ensure a complete and thorough understanding of the software's requirements to guide the development of a robust, secure, and user-friendly e-bidding platform.

In addition to outlining the user roles and functionalities, this document also aims to provide a comprehensive view of the software's integration with external systems, such as payment gateways and third-party APIs, for seamless transaction processing. The system will ensure high-level security features, such as encrypted transactions and secure user authentication, to safeguard sensitive user data. Moreover, the platform will be scalable, supporting future enhancements like mobile app integration and real-time bidding analytics. The document will also serve as a baseline for system validation and verification, ensuring that each functionality aligns with business goals and user needs.

1.2 Scope

The mini e-bidding system allows individuals to transact with other individuals in a secure and intuitive manner. Our system has an easy-to-use GUI so there is no confusion with any of the product’s utilities or features. The system consists of visitors, users, and super-users who would be the main actors. The visitor can view all listings, comment on them, and apply to be a user on the registration page. Users have more functionality, having the ability to create listings, choose which bidder to sell to, and bid on listings in specific listings pages, and withdraw/deposit funds or quit system in settings. Additionally, users can provide ratings and/or complaints (viewable by super-users) on other users who they have finished transacting with. Certain users who meet the criteria for VIP can gain perks, while others who have rating that falls under suspension criteria, will get suspended, in which they’ll see a suspension notice that requires them to pay $50. Super-users have access to all user applications (with ability to accept or deny user), complaints, and can see who’s suspended (with ability to unsuspend them) in their admin portal.

Beyond facilitating user transactions, the e-bidding system will also include advanced features such as automated bidding for users who wish to set predefined bidding limits. Users will also have access to a personalized dashboard, where they can track ongoing bids, view their transaction history, and manage their profile settings. The system will integrate with email and SMS services to notify users of important events, such as winning a bid, getting outbid, or receiving feedback. The admin interface for super-users will allow monitoring system health through analytics dashboards, ensuring optimal platform performance and user satisfaction.

1.3 Definitions, Acronyms, and Abbreviations

Super-user – An administrator who handles complaints, suspensions, and applications for becoming a user or quitting the system.
Visitor – Someone who can browse listings, comment on them, and apply to be a user.
User (VIP) – VIP stands for Very Important Person, and it's a user with perks such as a 10% discount on purchases and protection from suspension (though they can still have their VIP status removed).
GUI – Stands for Graphical User Interface, referring to the visual aspect of a software application.
API – Stands for Application Programming Interface, which allows software applications to communicate with each other. In this project, it allows the GUI to connect to the backend.
Bidding threshold – A limit set by users to automatically place bids on their behalf until this limit is reached.
Real-time bidding – A feature that updates the status of auctions in real-time as users place bids.
SSL (Secure Sockets Layer) – A standard security technology for establishing an encrypted link between the server and the browser to ensure data protection.
User dashboard – A section of the platform where users can manage their listings, bids, and account settings.

1.4 References

IBM. (2024, March 3). *Use-case template*. *https://www.ibm.com/docs/en/imdm/11.6?topic=cases-use-case-template*

1.5 Overview

The remainder of the document primarily consists of the overview of the project, consisting of the description and requirements. The description consists of use cases as well as assumptions we should make. Requirements are more specific, consisting of use case reports and supplementary requirements. Section 4 provides a space to have supporting information like use-case story boards or UI prototypes so that they can be referenced making the SRS easier to use.

1.6 Group Meeting Notes

6 November 2024

- Developed visuals of User's system screens

9 November 2024

- All Group Members Present
- Discuss about project scope and brainstorm ideas
- Created flowchart of events
- Distributed workload to members

2. Overall Description

The platform is designed with scalability and flexibility in mind, ensuring it can handle a growing user base and increasing demand for features like real-time bidding and automated notifications. The e-bidding system will operate on a microservices architecture, with different modules handling user management, bidding operations, and financial transactions. Users will be categorized into distinct profiles (e.g., VIP, suspended) that dynamically adjust based on their activity, performance, and feedback. The system will also employ user behavior analytics, tracking interactions to provide personalized recommendations and a customized experience for each user.

2.1 Use-Case Model Survey

Actors

1. Visitor – Along with browsing and commenting on listings, visitors are limited to view-only access until they apply to become a user.
2. User – Besides listing and bidding on items or services, users can also manage their account settings, including their profile information, transaction history, and payment methods.
3. Super-user – Administrators not only manage complaints and interactions but can also handle account suspensions, approve or deny user status changes, and monitor system abuse.

Use Cases

1. View Listings – Listings should be searchable by category, location, and price, allowing users and visitors to filter through results based on their needs. There should also be pagination to prevent overload.
2. Add Comment to Listing – Comments can be nested to allow users and visitors to reply to other comments. There should also be a mechanism for reporting inappropriate comments.
3. Create Listing – Users will have a listing form with categories for the item type, pricing structure, and other relevant details, such as condition or availability dates for services.
4. Apply to be User – Allow visitors to submit an application to be a User
5. Deposit / Withdraw Money – Allow users to add funds to the platform or retrieve back their funds
6. Bid on Listing – The bidding interface will show current highest bids, bidder history, and the time remaining on the bid. Users will be notified if they have been outbid.
7. View Complaints – Allow Super-users to view all complaints
8. Apply to quit out of system – Allow Users to apply to quit out of the system
9. Accept/Deny Application to Quit System – Allow Super-users to accept or deny a user's application to leave the system
10. Accept/Deny Application to be User – Allow Super-users to accept or deny a visitor's application to be a User.
11. Write Complaint – Allow users to write a complaint about another user after a transaction
12. Give Rating – Allow users who were involved in a transaction to submit a rating for each other
13. Accept Bidder – Allow user who listed an item or service to select a bidder (if any) to sell / rent to
14. Unsuspend User – Allow Super-user to unsuspend a suspended user
15. Pay Fine – Allow suspended users to pay $50 fine to get unsuspended
16. Login – Allow Users and Super-users to login
17. Accept Bidder – The system should notify users when bids are accepted or rejected, along with the option to send a confirmation message to the winning bidder.

2.2 Assumptions and Dependencies

No Direct Connection to Bank Account or Funds

- We will omit setting up connections to APIs that facilitate connections to bank accounts or credit cards and assume withdrawal/deposits work with an already connected source of funds.

Web Application

- GUI will be web based so that it can be easily accessible from mobile to desktop through their dedicated browser.

Stable Internet Connection

- To connect to the Web app GUI and to partake in bidding/selling the user should have a stable connection on the internet to be able to access and execute all application functions

3. Specific Requirements

3.1 Use-Case Reports

Use Case ID	**UC-001**
Use Case Title	Visitor application to be a User
Subject area	Handling Visitor and User
Actor(s)	Super-user, Visitor
Use case overview	Visitor registering to be a User with application and Super User deciding to accept or reject application
Preconditions	The visitor is not a user and has not been kicked out of the system.
Termination Outcome	- The visitor submits a successful application, and Super User grants the visitor the User role. (Happy Case) - The visitor submits an unsuccessful application, and Super User denies visitor the User role.
Usability index	- Priority: High - Frequency: As needed

Use Case ID	**UC-002**
Use Case Title	User login
Subject area	User account access
Actor(s)	User
Use case overview	User logging in to system
Preconditions	The visitor is not a user and has not been kicked out of the system.
Termination Outcome	- The user enters the proper credentials to access their account - The user enters the incorrect credentials and does not gain access to their account - User must reset password to regain access to account
Usability index	- Priority: High - Frequency: As needed

Use Case ID	**UC-003**
Use Case Title	User logout

Subject area	User account access
Actor(s)	User
Use case overview	User logging out of system
Preconditions	The visitor is not a user and has not been kicked out of the system.
Termination Outcome	- The user successfully logs out of their account and must log in once again to retrieve relevant data
Usability index	- Priority: High - Frequency: As needed

Use Case ID	**UC-004**
Use Case Title	User application to quit system
Subject area	Handling Visitor and User
Actor(s)	Super-user, User
Use case overview	Users apply to quit the system and lose their user role within the system.
Preconditions	The visitor is a user and has not been kicked out of the system.
Termination Outcome	- The visitor submits a successful application and Super removes the User from the system. (Happy Case) - The visitor submits an unsuccessful application, and Super User denies User from exiting the system.
Usability index	- Priority: High - Frequency: As needed

Use Case ID	**UC-005**
Use Case Title	Unsuspend user
Subject area	Handling Visitor and User
Actor(s)	Super-user, User
Use case overview	User is suspended due to neglectful or too generous ratings and must either pay a fine to rejoin or must be unsuspended by Super-user
Preconditions	The User is suspended
Termination Outcome	- The user pays the fines and gets the account unsuspended - A Super User unsuspends the account
Usability index	- Priority: High - Frequency: As needed

Use Case ID	**UC-006**

Use Case Title	Unsuspend user
Subject area	Handling Visitor and User
Actor(s)	Super-user, User
Use case overview	The user is suspended due to neglectful or too generous average ratings and must either pay a fine to rejoin or must be unsuspended by the Super-user
Preconditions	The User is suspended
Termination Outcome	- The user pays the fines and gets the account unsuspended - A Super User unsuspends the account
Usability index	- Priority: High Frequency: As needed

Use Case ID	**UC-007**
Use Case Title	Create Listing
Subject area	User actions
Actor(s)	User
Use case overview	The user creates a listing that other Users can bid on.
Preconditions	-
Termination Outcome	- User successfully lists item/service to website for Visitors and Users to see and only for Users to bid on based on respective asking price and/or price ranges.
Usability index	- Priority: High - Frequency: As needed

Use Case ID	**UC-008**
Use Case Title	Updating account balance
Subject area	User actions
Actor(s)	User
Use case overview	The user updates their account balance by either depositing or withdrawing
Preconditions	The User is logged in and not suspended or has been removed from the system
Termination Outcome	- User successfully adds funds to their account - User successfully removes funds from their account - User is unable to remove or add from/to external bank account
Usability index	- Priority: High - Frequency: As needed

Use Case ID	**UC-009**
Use Case Title	Updating account balance
Subject area	User actions
Actor(s)	User
Use case overview	The user updates their account balance by either depositing or withdrawing
Preconditions	The User is logged in and not suspended or has been removed from the system
Termination Outcome	- User successfully adds funds to their account - User successfully removes funds from their account - User is unable to remove or add from/to external bank account
Usability index	- Priority: High - Frequency: As needed

Use Case ID	**UC-010**
Use Case Title	Bid on item/service
Subject area	User actions
Actor(s)	User
Use case overview	The user is bidding on an item/service listed by another user
Preconditions	The User is logged in and not suspended or has been removed from the system
Termination Outcome	- User successfully bids on item/service by specifying a pay range - The bid fails due to the item having been bought or amount allocated lower than the asking price
Usability index	- Priority: High - Frequency: As needed

Use Case ID	**UC-011**
Use Case Title	Give Rating
Subject area	User actions
Actor(s)	User
Use case overview	The user has won a bid and wants to leave a rating
Preconditions	The User is logged in and not suspended or has been removed from the system and has successfully won a bid for an item/service
Termination Outcome	- Leaves an anonymous rating for the User who listed the item/service
Usability index	- Priority: High

	- Frequency: As needed

Use Case ID	**UC-012**
Use Case Title	Accept Bid
Subject area	User actions
Actor(s)	User
Use case overview	The user who put up the item/service to be bid on has found a User that has offered a good price and so they would like to accept the price proposed.
Preconditions	The User is logged in and not suspended or has been removed from the system and has successfully listed an item/service to be bid
Termination Outcome	- The User who listed the bid accepts the bid placed by the buying/renting user and once the User who placed the bid also accepts the bid has ended and the item is taken down from the listings
Usability index	- Priority: High - Frequency: As needed

Use Case ID	**UC-013**
Use Case Title	Viewing and commenting on listing
Subject area	-
Actor(s)	User, Visitor
Use case overview	The User/visitor is viewing the listings from other Users and wants to leave a comment on the listing.
Preconditions	The User is logged in and not suspended or has been removed from the system. The Visitor is not a User and only contains the ability to view and comment on listing.
Termination Outcome	Users/Visitors view the current state of any listings created by users in the system and leave a comment on them
Usability index	- Priority: Medium - Frequency: As needed

Use Case ID	**UC-014**
Use Case Title	Viewing complaints
Subject area	-
Actor(s)	Super User, User
Use case overview	The user has bought an item/service and would like to complain about the experience.

Preconditions	The User is logged in and not suspended or has been removed from the system, and has made a purchase of the respective item for which they are complaining about
Termination Outcome	- The complaint is valid and the User who listed the item/service to be bid receives a complaint strike and becomes ineligible to be a VIP User - The complaint is invalid and so it is dropped, and no strike is given to the author who listed the item/service
Usability index	- Priority: Medium Frequency: As needed

Use Case ID	**UC-015**
Use Case Title	Notifications System
Subject area	-
Actor(s)	Super User, User
Use case overview	Users should be notified when a key action occurs on their listing or bids. Notifications are delivered in-app, via email, or SMS, based on user preferences.
Preconditions	The User is logged in and not suspended or has been removed from the system, and the User has posted a listing or placed a bid on a listing
Termination Outcome	- Successful notifications: The system successfully delivers the notification through the selected channel(s). The User is informed of the event (bid placed, comment, bid won). - Unsuccessful notifications: The system fails to deliver the notification (e.g., due to incorrect email/SMS details). The system logs the failure, and the User can see the notification in-app.
Usability index	- Priority: Medium Frequency: As needed

3.2 Supplementary Requirements

Live Bidding Session – VIP users can host time sensitive auctions, where only VIPs can participate
Search Bar – Query item or service you're interested in or want to sell to.
Two-Factor Authentication – Add the option for users to enable two-factor authentication for added security, especially for actions like deposits, withdrawals, and bidding on high-value items.

4. Supporting Information

4.1 Index

- Actors
- Use Cases
- User Actions
- Preconditions
- Termination Outcomes
- Usability Index
- Supplementary Requirements

4.2 Appendices

A.1 Actor Descriptions

- Visitor: A non-registered individual who can browse listings and comment on them.
- User: A registered member who can create listings, bid on items, and manage their account.
- Administrator: An elevated user responsible for overseeing user activity and managing system complaints.

A.2 Use Case Overview

- Use Cases: Browse Listings: Allows all users and visitors to view available listings for items or services.
- Comment on Listings: Enables users and visitors to add comments to listings for feedback or inquiries.
- List Items/Services: Users can create listings for items or services they wish to sell, rent, or seek.
- User Registration: Visitors can submit applications to register as users.
- Account Management: Users can add or withdraw funds, manage their profiles, and handle account-related issues.
- Bidding Process: Users can place bids on items or services listed by others.
- View Feedback: Super-users can access and review all user complaints and feedback.
- Account Deactivation: Users can apply to remove themselves from the platform, pending administrator approval.
- Application Review: Administrators can approve or reject user registration and deactivation requests.
- Transaction Feedback: Users can rate their transaction experiences with others.
- Accept Bids: Sellers can select winning bids from potential buyers.
- User Suspension Management: Administrators can suspend or reinstate users based on activity.

A.3 System Constraints

- No Direct Financial Transactions: The system does not handle direct bank connections for payments or withdrawals, assuming that users will manage funds externally.
- Web Accessibility: The application will be developed as a web-based platform for compatibility across devices.
- Internet Connectivity Requirement: Users must have a reliable internet connection to access the application’s functionalities.
- B. Functional Requirements
- The following table outlines functional requirements based on use cases:

Use Case ID, Title, Functional Requirements

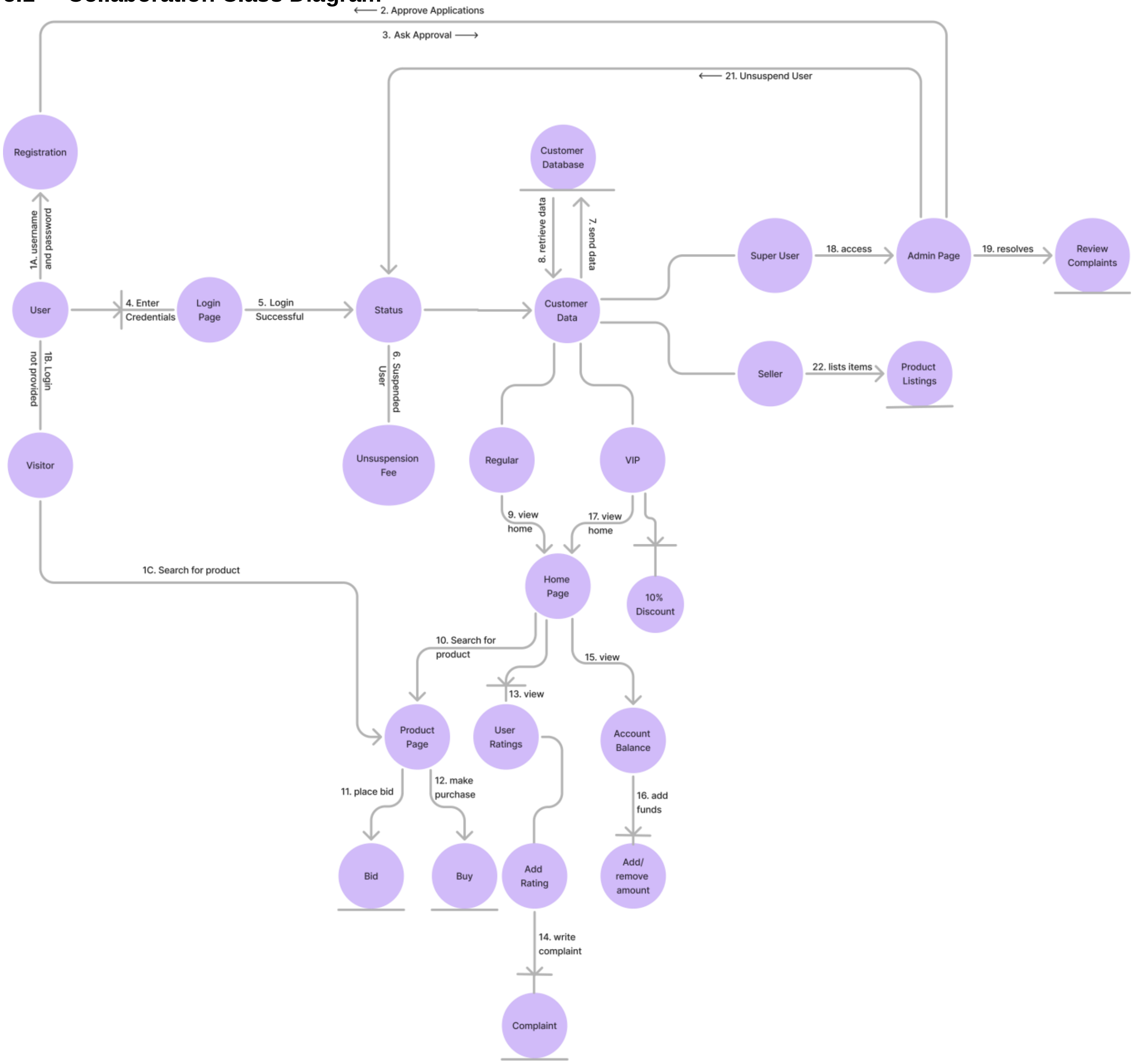
- UC-001, User Registration: Users must provide necessary details to create an account.
- UC-002, User Login: Users must enter valid credentials to access their accounts.
- UC-003, User Logout: Users must be able to log out securely from their accounts.
- UC-004, Apply to Quit: Users can submit requests to delete their accounts, which must be reviewed by an administrator.
- UC-005, User Suspension: Administrators can suspend users based on misconduct or violation of terms.
- UC-006, Create Listing: Users must be able to create listings for items or services.
- UC-007, Update Account Balance: Users can deposit or withdraw funds from their accounts.
- UC-008, Place Bid: Users can bid on listed items within specified limits.
- UC-009, Rate Transactions: Users can provide feedback on their transaction experiences.
- UC-010, Accept Bids: Sellers can approve bids made by potential buyers.

C. Non-Functional Requirements

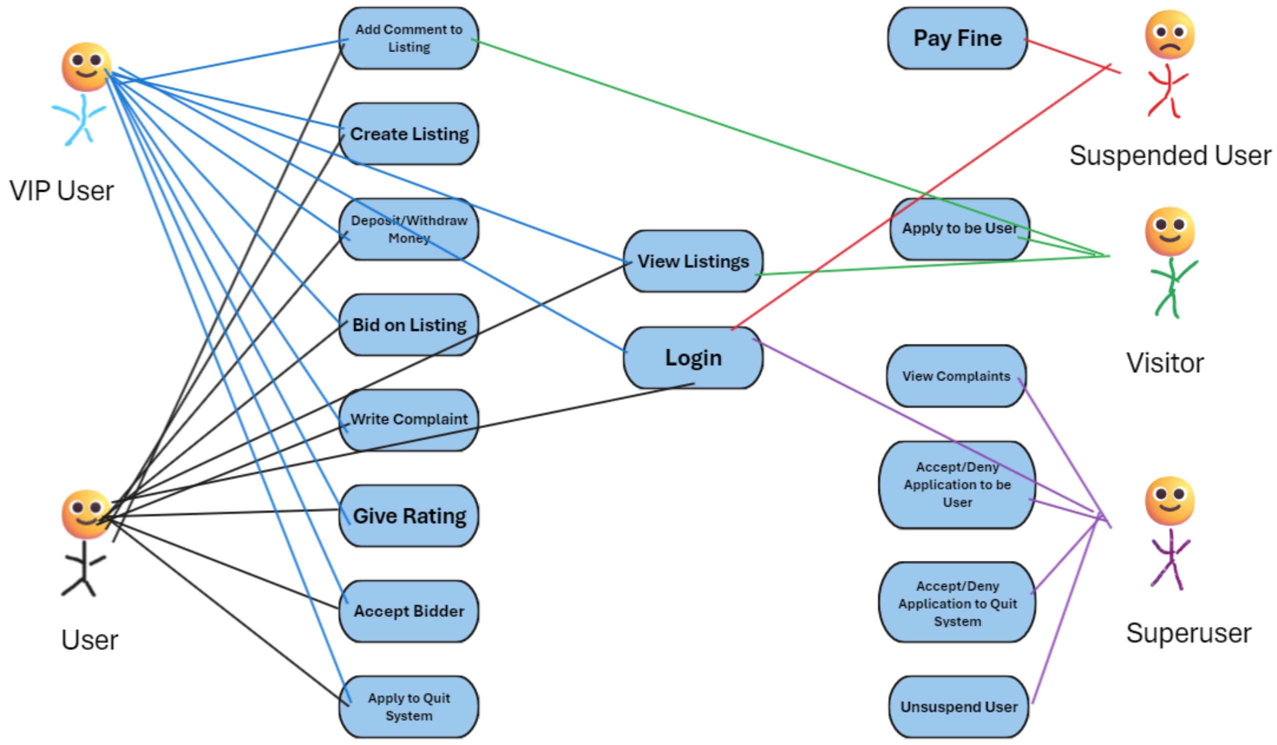
- Performance: The system should be able to handle up to 1000 concurrent users without degradation of performance.
- Security: User data must be encrypted, and the application should adhere to best practices for secure data handling.
- Usability: The interface should be user-friendly, with intuitive navigation and clear instructions.

5. Design Diagrams and Prototype

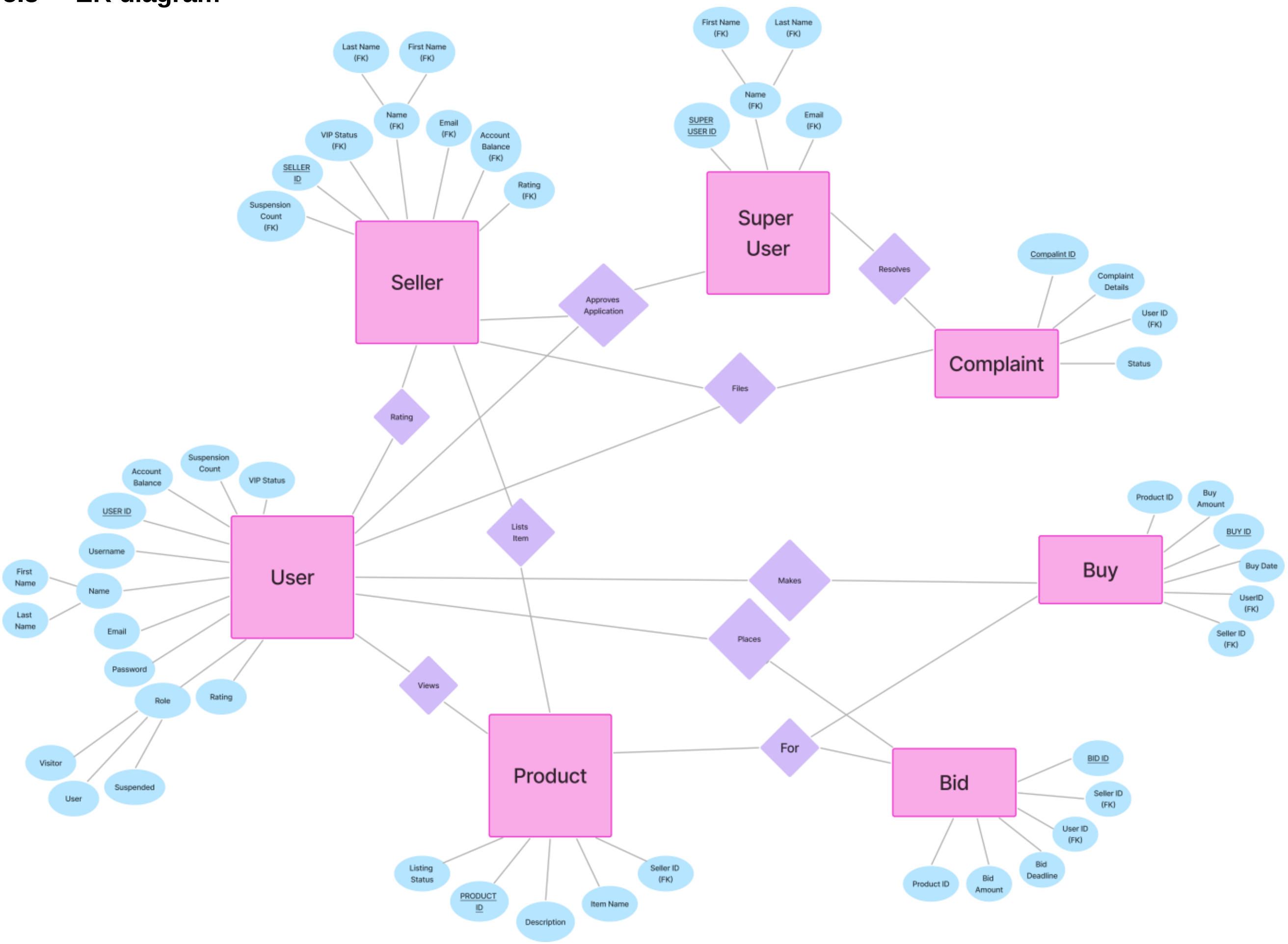
5.1 Collaboration Class Diagram



5.2 Use Cases

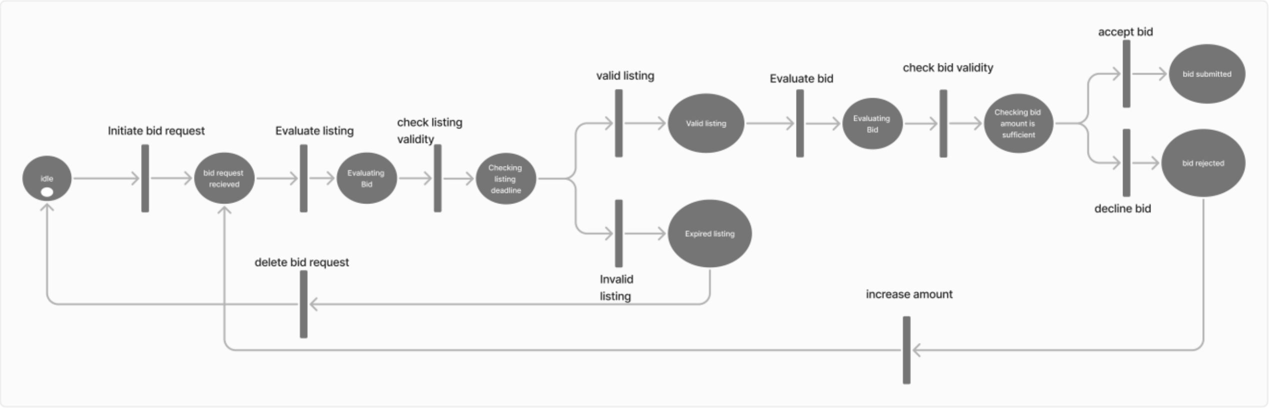

VIP User
User
Add Comment to Listing
Create Listing
Deposit/Withdraw Money
Bid on Listing
Write Complaint
Give Rating
Accept Bidder
Apply to Quit System
View Listings
Login
Pay Fine
Apply to be User
View Complaints
Accept/Deny Application to be User
Accept/Deny Application to Quit System
Unsuspend User
Suspended User
Visitor
Superuser

5.3 ER diagram

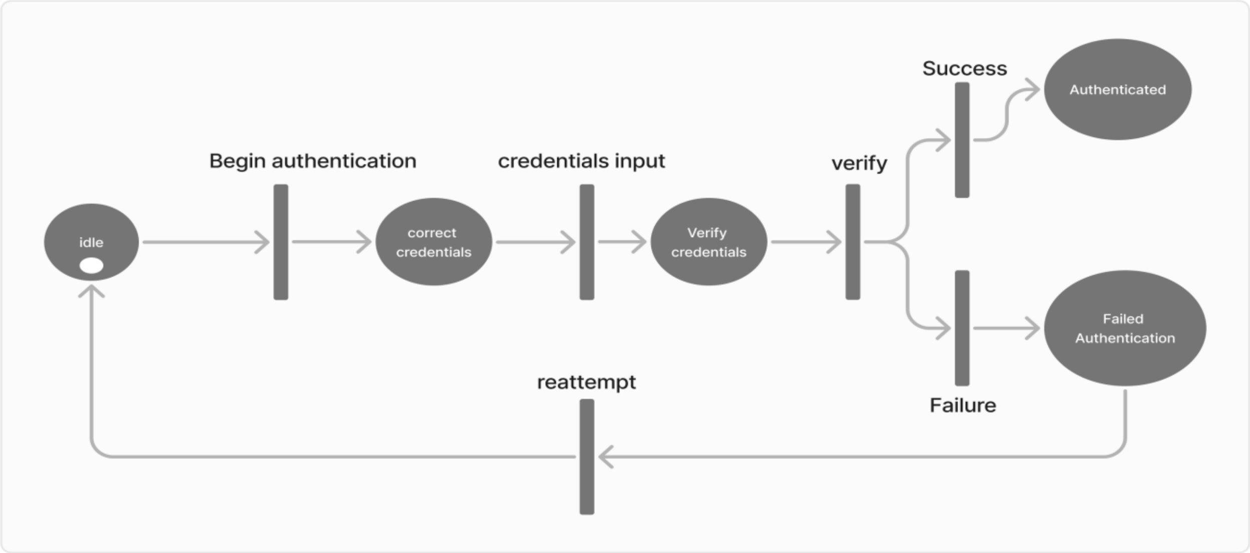


5.4 Petri Net

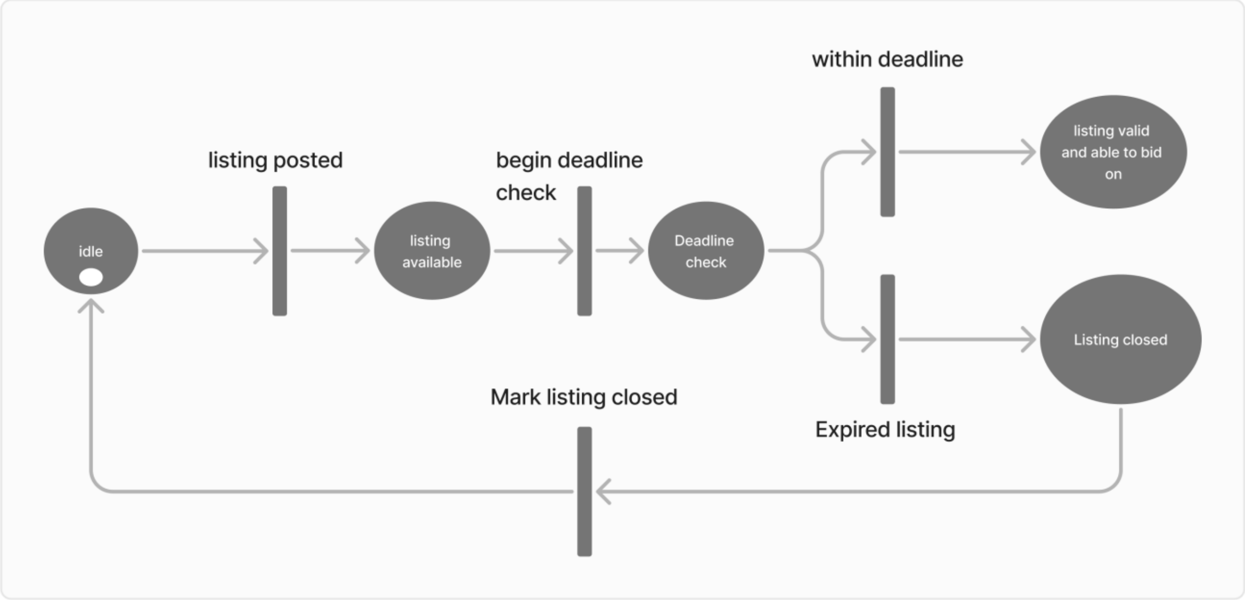
Bidding



User Authentication

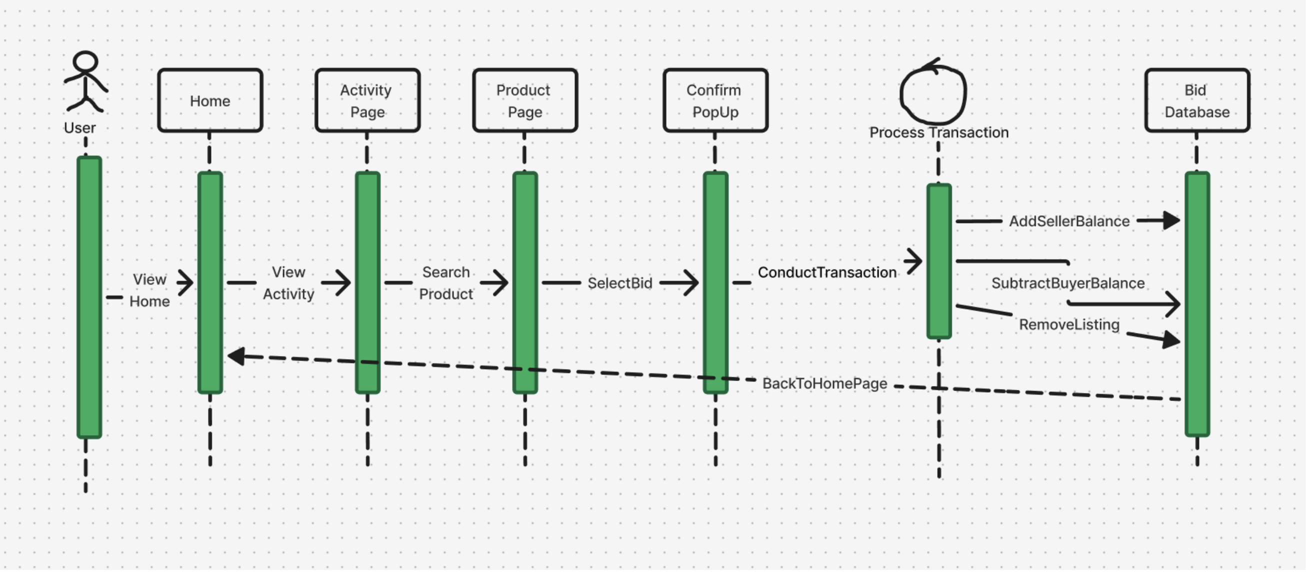


Listing Deadline

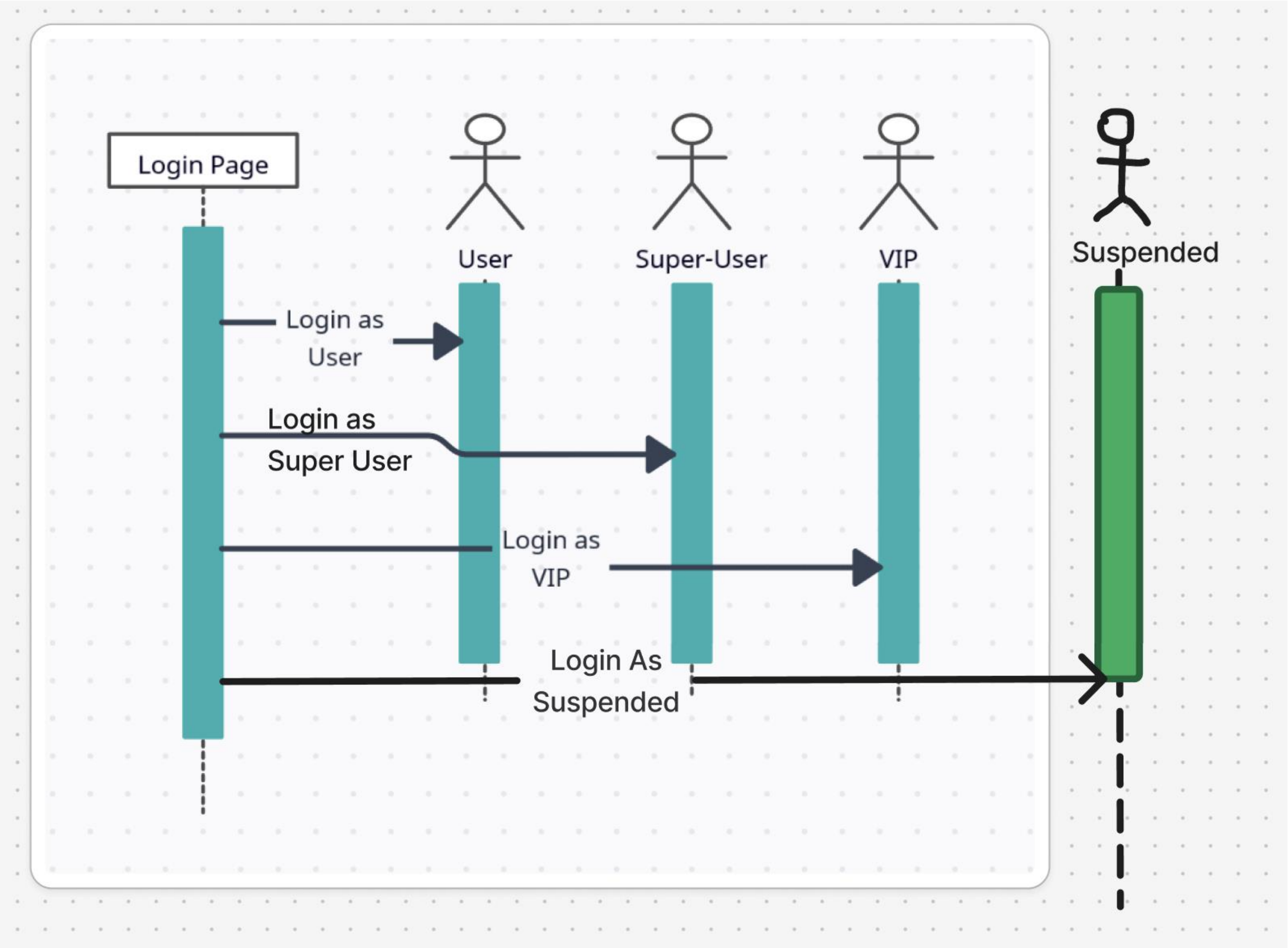


5.5 Sequence Class Diagram

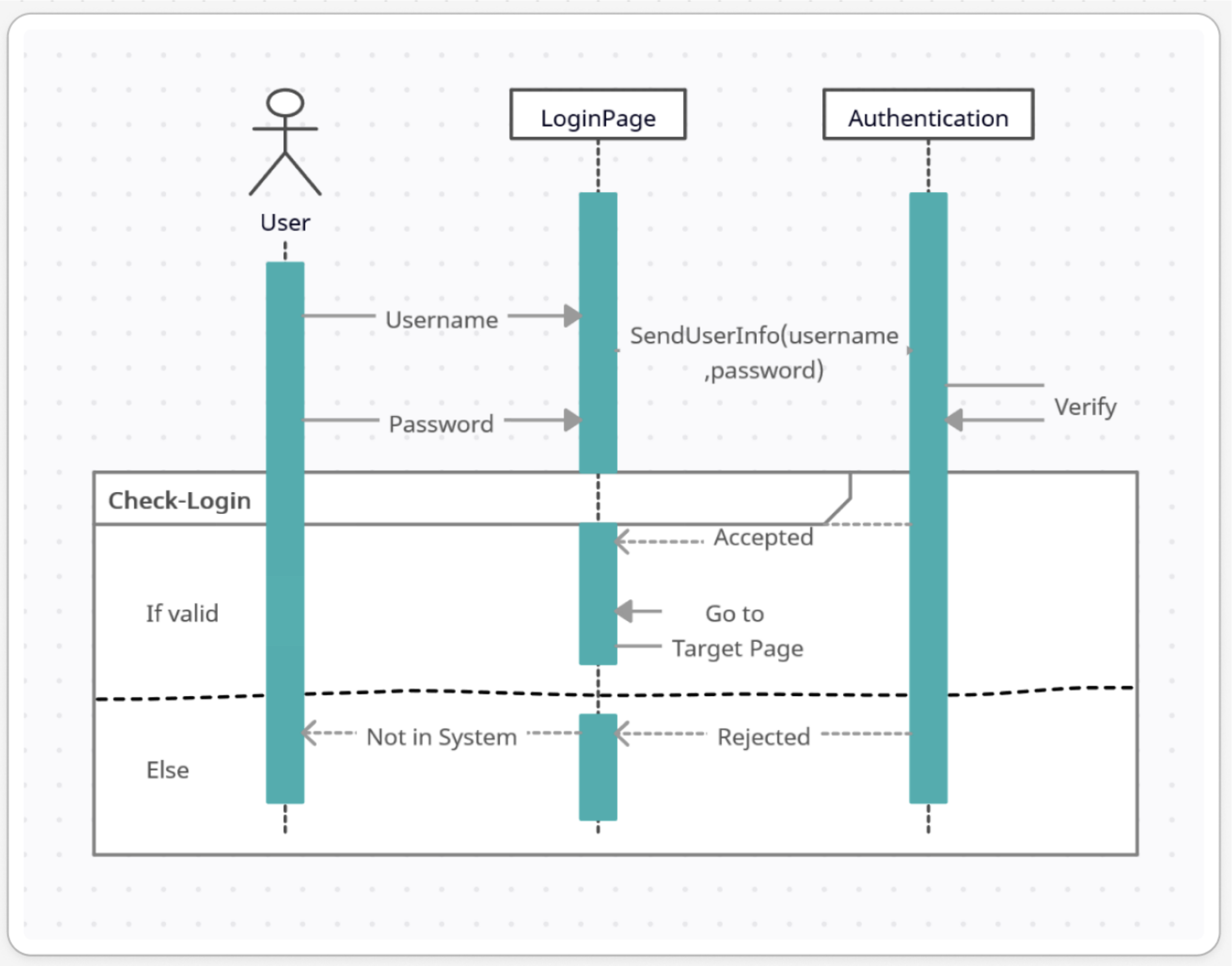
Buyer Accepting Bid Sequence Class Diagram



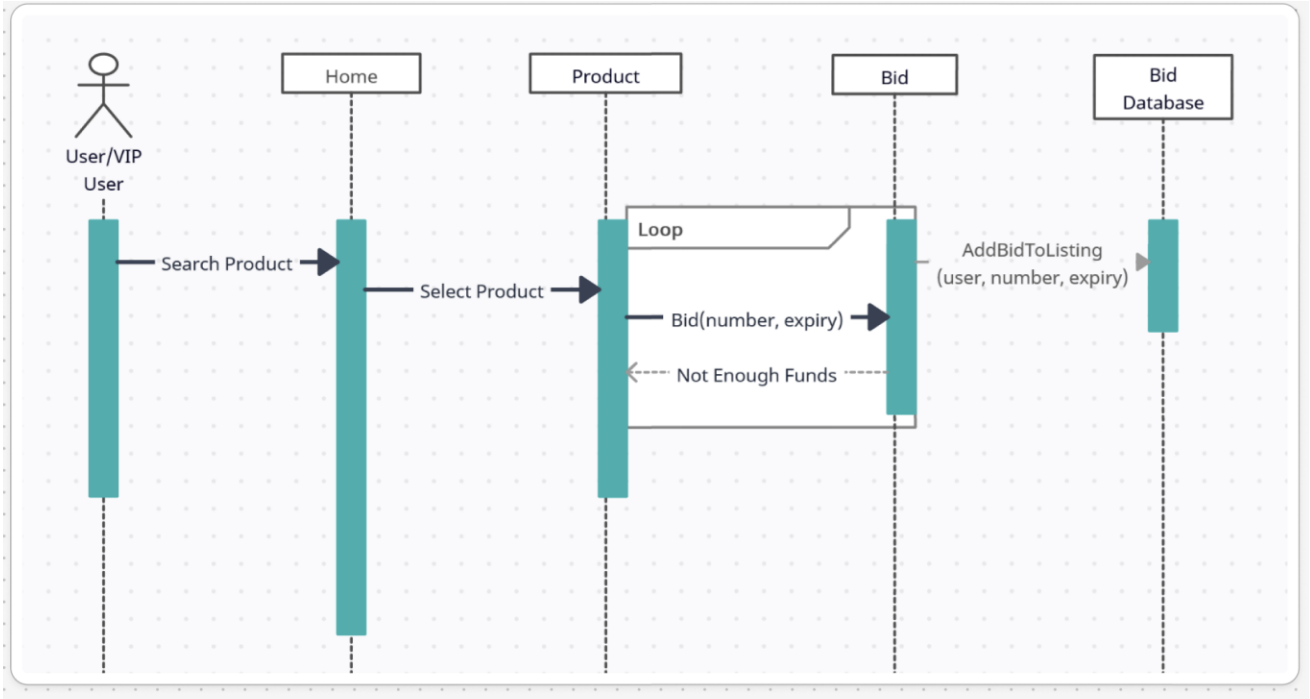
Login Sequence Class Diagram



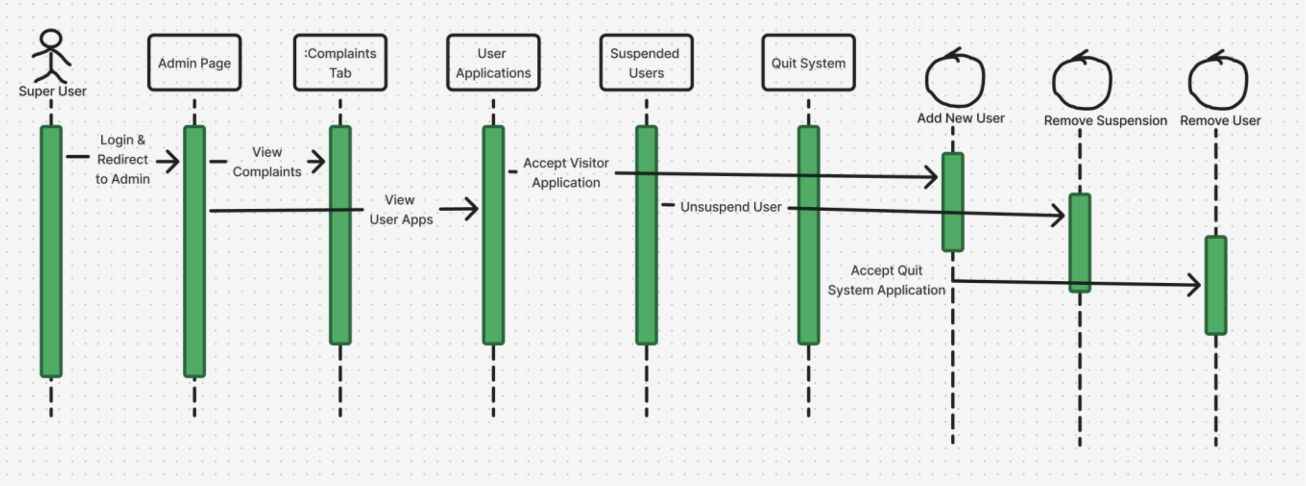
Authentication Sequence Class Diagram



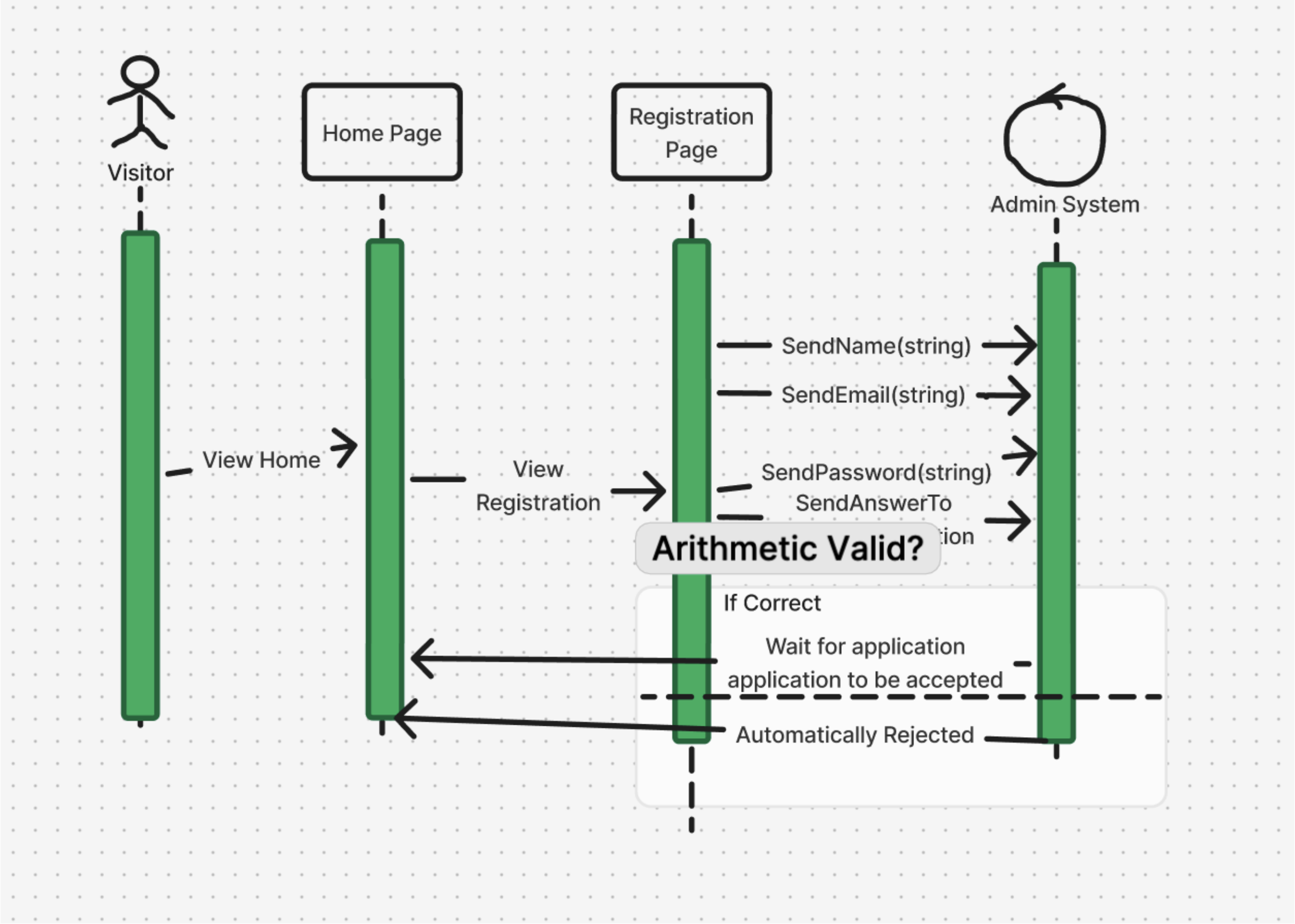
Bidding Sequence Class Diagram



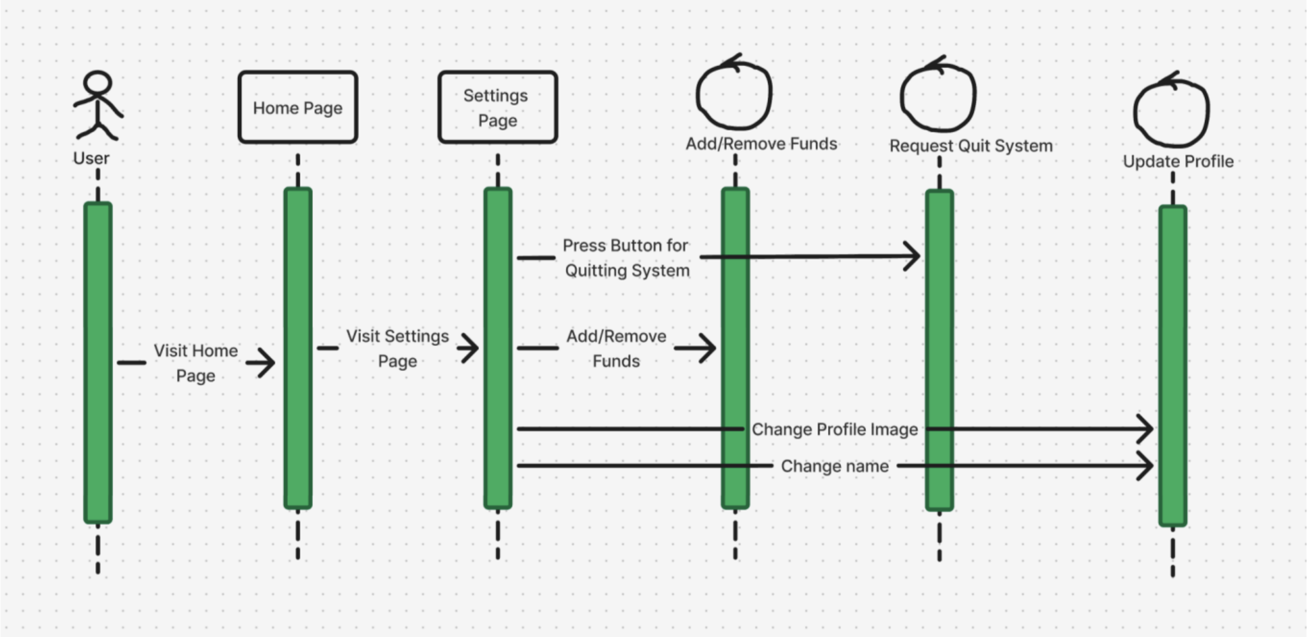
Super User Activity Sequence Class Diagram



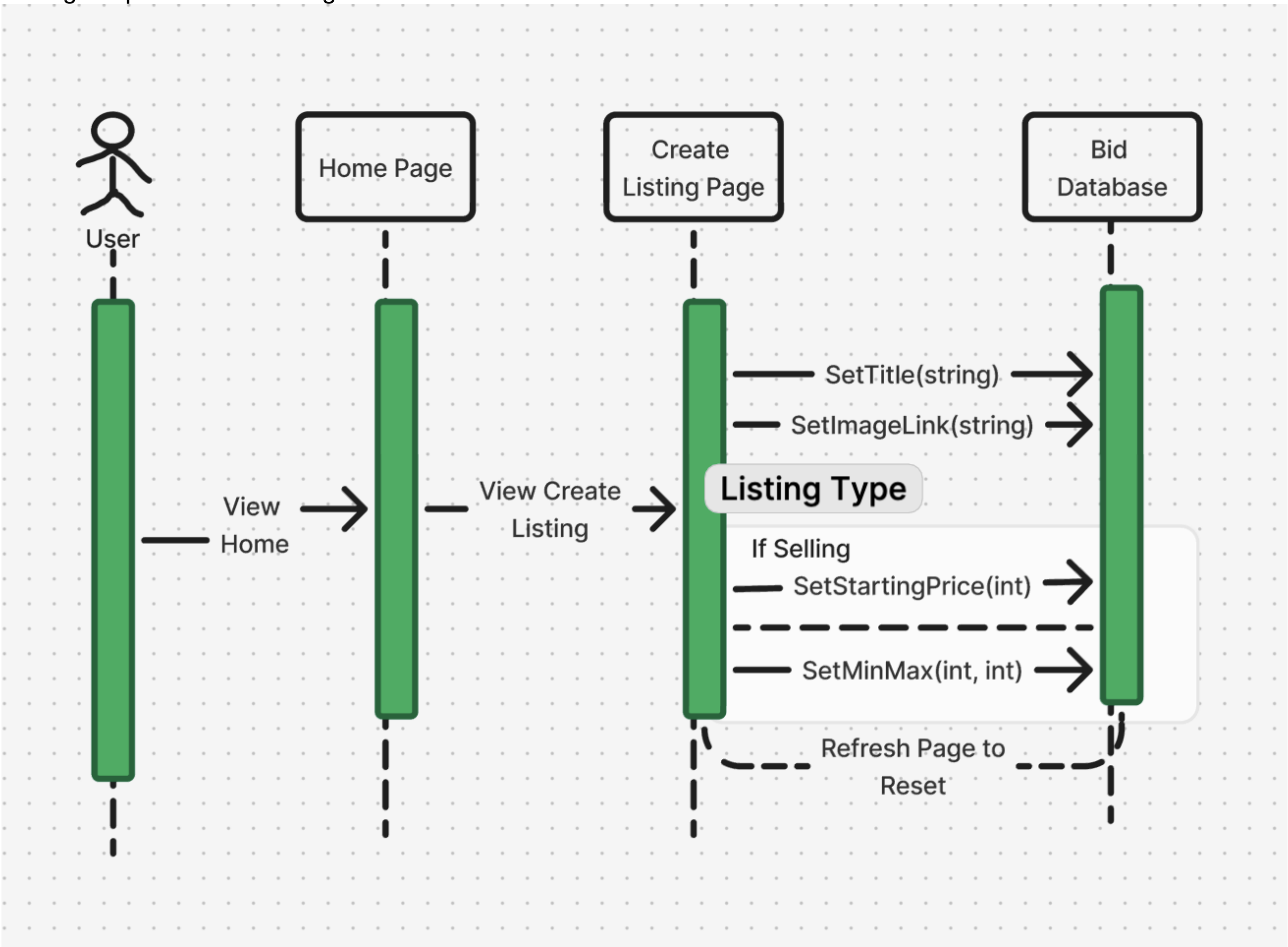
User Registration Sequence Class Diagram



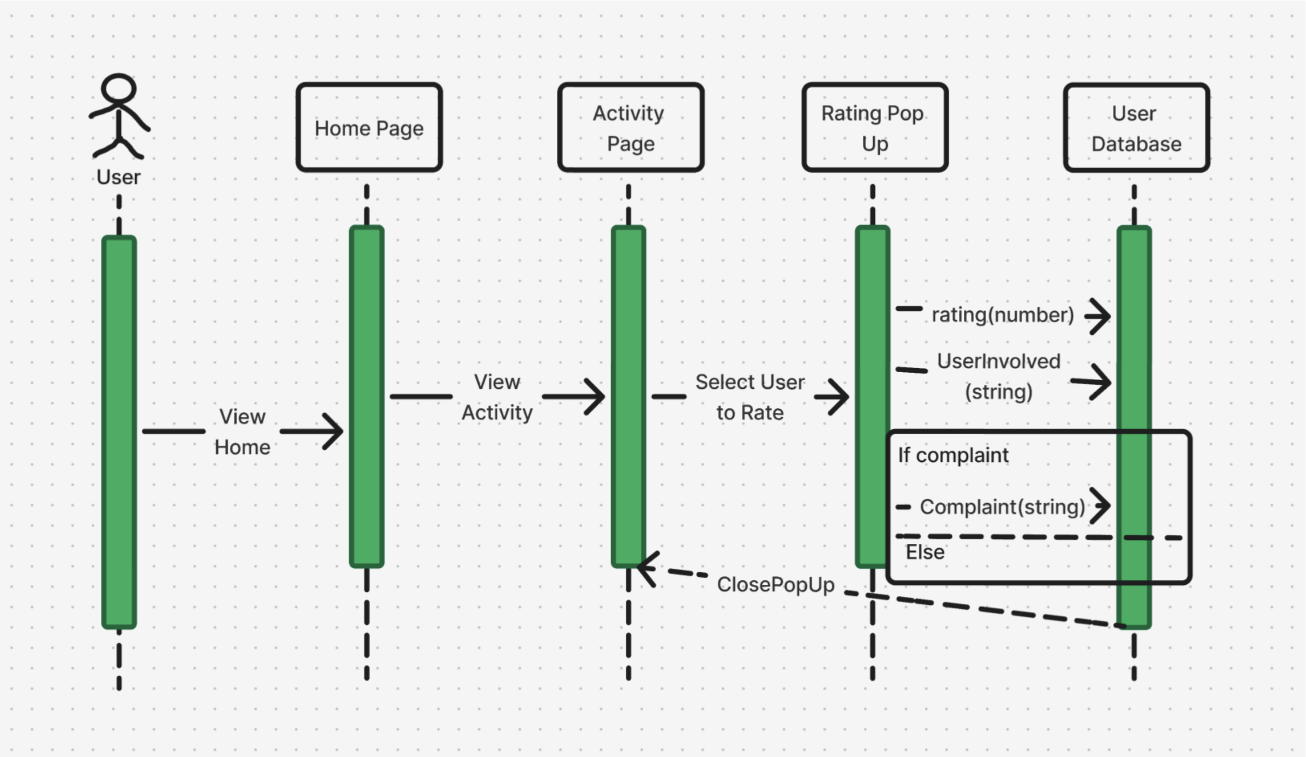
Settings Sequence Class Diagram



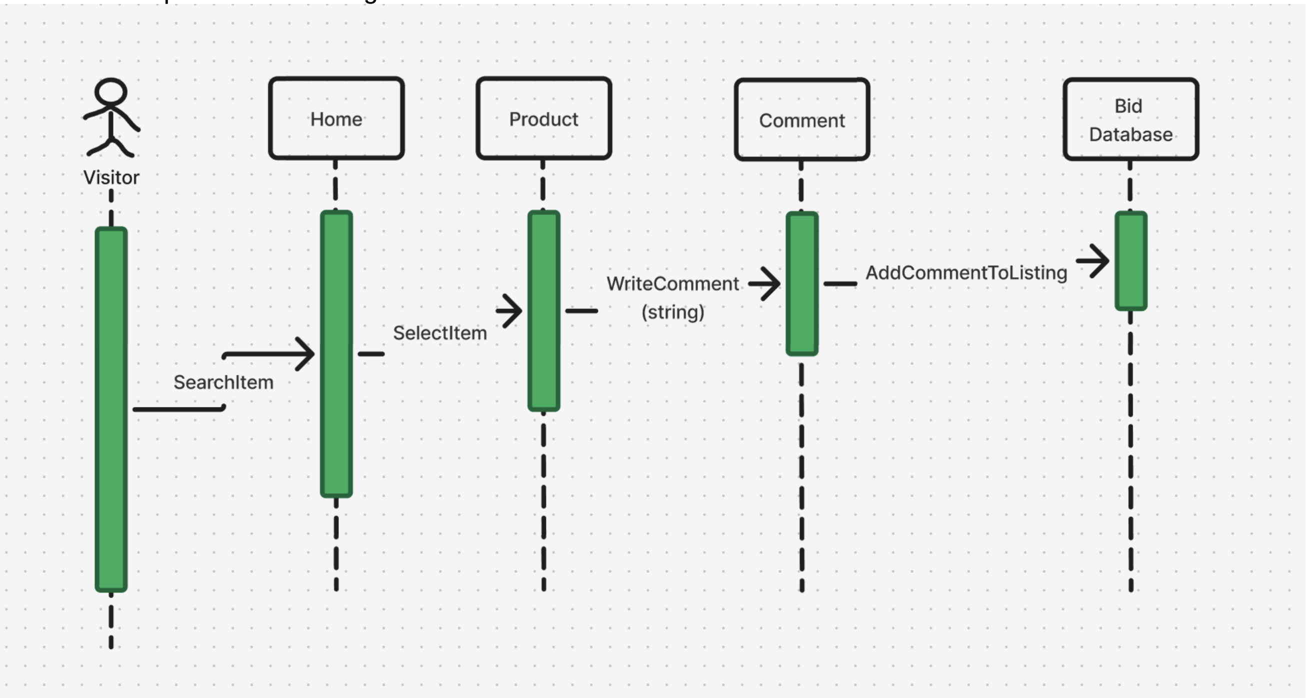
Listing Sequence Class Diagram



Rating/Complain Sequence Class Diagram

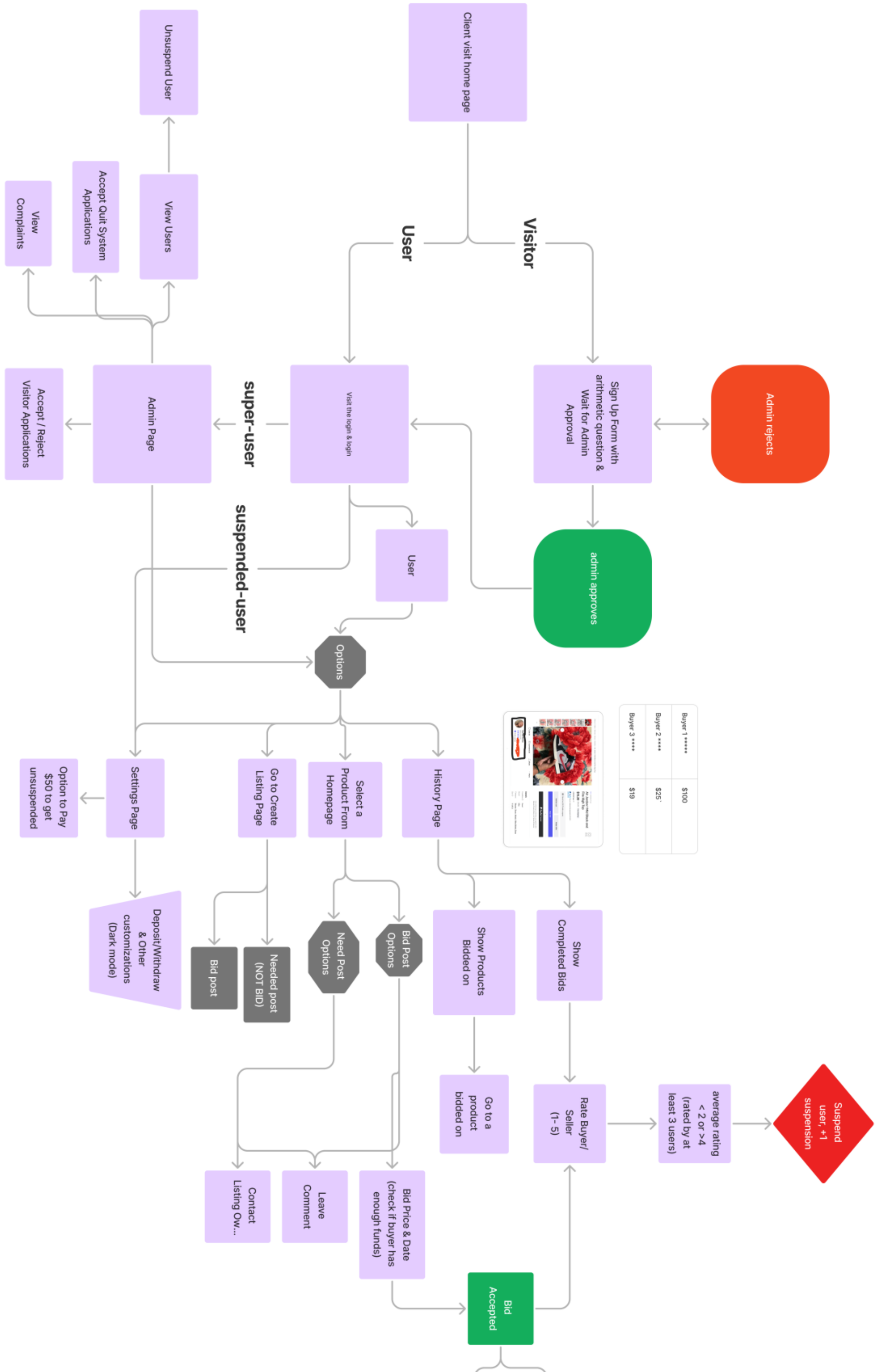


Comment Sequence Class Diagram



5.6 Flow chart diagram

End to End app logic


Client visit home page
Visitor
User
Sign Up Form with arithmetic question & Wait for Admin Approval
Admin rejects
admin approves
Visit the login & login
super-user
suspended-user
User
Admin Page
Accept / Reject Visitor Applications
View Users
Unsuspend User
Accept Quit System Applications
View Complaints
Options
History Page
Select a Product From Homepage
Go to Create Listing Page
Settings Page
Option to Pay $50 to get unsuspended
Deposit/Withdraw & Other customizations (Dark mode)
Bid post
Needed post (NOT BID)
Need Post Options
Bid Post Options
Show Products Bidded on
Show Completed Bids
Go to a product bidded on
Rate Buyer/ Seller (1- 5)
average rating < 2 or >4 (rated by at least 3 users)
Suspend user, +1 suspension
Contact Listing Ow...
Leave Comment
Bid Price & Date (check if buyer has enough funds)
Bid Accepted
Buyer 1 *****
$100
Buyer 2 ****
$25
Buyer 3 ****
$19

6. Detailed Design and Prototype

user_sign_up:

- Precondition: User fills out a sign form with username, password and solves an arithmetic question
- Postcondition: User account is created and waits for admin approval
- Pseudo Code:

```
function user_sign_up (username, password, arithmeticAnswer):
   if validateArithmetic(arithmeticAnswer):
      createUser(username, password)
      notifyAdminForApproval(username)
```

user_login:

- Precondition: User enters their username and password
- Postcondition: User logs in if the credentials are valid and not suspended
- Pseudo Code:

```
function user_login(username, password):
   if isValidCredentials(username, password) and not user.isBanned():
      logUserIn()
   else:
      notifyUser(username, "Login failed or account is banned.")
```

admin_approval:

- Precondition: Admin receives a notification for a pending user
- Postcondition: User account is approved, giving access to the app
- Pseudo Code:

```
function admin_approve(username):
   user = findUser(username)
   if user and user.isPendingApproval:
      user.approve()
      notifyUser(username, "Your account has been approved.")
```

admin_rejection:

- Precondition: Admin finds a pending user application
- Postcondition: User application is rejected, and user is notified
- Pseudo Code:

```
function admin_rejection(username):
   user = findUser(username)
   if user and user.isPendingApproval:
      user.reject()
 notifyUser(username, "Your account application was rejected.")
```

suspension_check:

- Precondition: User attempts login with more than 3 suspensions or violates average rating conditions
- Postcondition: User is banned or suspended accordingly
- Pseudo Code:

```
function suspension_check(username):
   user = findUser(username)
```

```
if user.suspensions >= 3:
    user.ban()
    notifyUser(username, "Your account has been banned.")
elif user.rating < 2 or user.rating > 4 and user.totalRatings >= 3:
    user.suspend()
    notifyUser(username, "You are temporarily suspended due to ratings.")
```

unsuspension_option:

- Precondition: If user is suspended, they can pay $50 to get unsuspended
- Postcondition: User suspension will be lifted
- Pseudo Code:

```
function unsuspension_option(username):
    user = findUser(username)
    if user.isSuspended() and user.pay(50):
        user.unsuspend()
        notifyUser(username, "You are now unsuspended.")
```

unsuspend_user:

- Precondition: Admin decides to lift a suspension
- Postcondition: User’s suspension is removed
- Pseudo Code:

```
function unsuspend_user (username):
    user = findUser(username)
    if user.isSuspended() and user.pay(50):
        user.unsuspend()
        notifyUser(username, "You are now unsuspended.")
```

visitor_application:

- Precondition: Admin reviews visitor application
- Postcondition: Visitor is granted or denied access to the platform
- Pseudo Code:

vip_status:

- Preconditon: User has $5000, more than 5 ratings and no complaints
- Postcondition: User is given the Vip role
- Pseudo Code:

```
function vip_status(username):
    user = findUser(username)
    if user.balance > 5000 and user.ratings >= 5 and user.complaints == 0:
        user.setRole("VIP")
```

have_complaint:

- Precondition: A compliant is filled against the user
- Postcondition: User’s complaint counts increases and may lose VIP status
- Pseudo Code:

```
function have_complaint (username):
    user = findUser(username)
    user.complaints += 1
    if user.isVIP():
        user.setRole("Regular")
```

view_complaints:

- Precondition: Admin accesses complaint records

- Postcondition: Admin reviews list of complaints for action
- Pseudo Code:

```
function view_complaints():
    return database.getComplaints()
```

rate_user:

- Precondition: Transaction completed; users can rate each other
- Postcondition: Rating is added to the user’s profile
- Pseudo Code:

```
function rate_user(rater, ratee, rating):
    if rating >= 1 and rating <= 5:
        ratee.addRating(rating)
        notifyUser(rater, "Rating submitted.")
```

create_bid:

- Precondition: User selects a product to place a bid on and specifies bid amount and date
- Postcondition: Bid is created if user has sufficient funds
- Pseudo Code:

```
function create_bid(user, product, bidAmount, bidDate):
    if user.balance >= bidAmount:
        product.addBid(user, bidAmount, bidDate)
```

accept_bid:

- Precondition: Listing owner accepts a bid
- Postcondition: Bid is accepted, funds are transferred, and listing is removed
- Pseudo Code:

```
function accept_bid(listingOwner, bid):
    if bid.isValid():
        transferFunds(bid.buyer, listingOwner, bid.amount)
        bid.product.removeListing()
```

products_bidded:

- Precondition: User navigates to their profile
- Postcondition: Displays all products the user has bid on
- Pseudo Code:

```
function products_bidded (user):
    return user.getBiddedProducts()
```

deposit:

- Precondition: User initiates a deposit
- Postcondition: Balance is updated accordingly
- Pseudo Code:

```
function deposit(user, amount):
    user.balance += amount
    notifyUser(user, f"${amount} has been deposited into your account.")
```

withdraw:

- Precondition: User initiates a withdrawal
- Postcondition: Balance is updated accordingly
- Pseudo Code:

```
function withdraw(user, amount):
    if user.balance >= amount:
        user.balance -= amount
        notifyUser(user, f"${amount} has been withdrawn from your account.")
```

create_listing:

- Precondition: User has appropriate permissions and creates a new product listing
- Postcondition: Listing is added to the marketplace
- Pseudo Code:

```
function create_listing(user, product):
    if user.canCreateListing():
        marketplace.addListing(product)
```

contact_owner:

- Precondition: User wants to contact the owner of a listing
- Postcondition: Message is sent to the listing owner
- Pseudo Code:

```
function contact_owner(user, listingOwner, message):
            sendMessage(user, listingOwner, message)
```

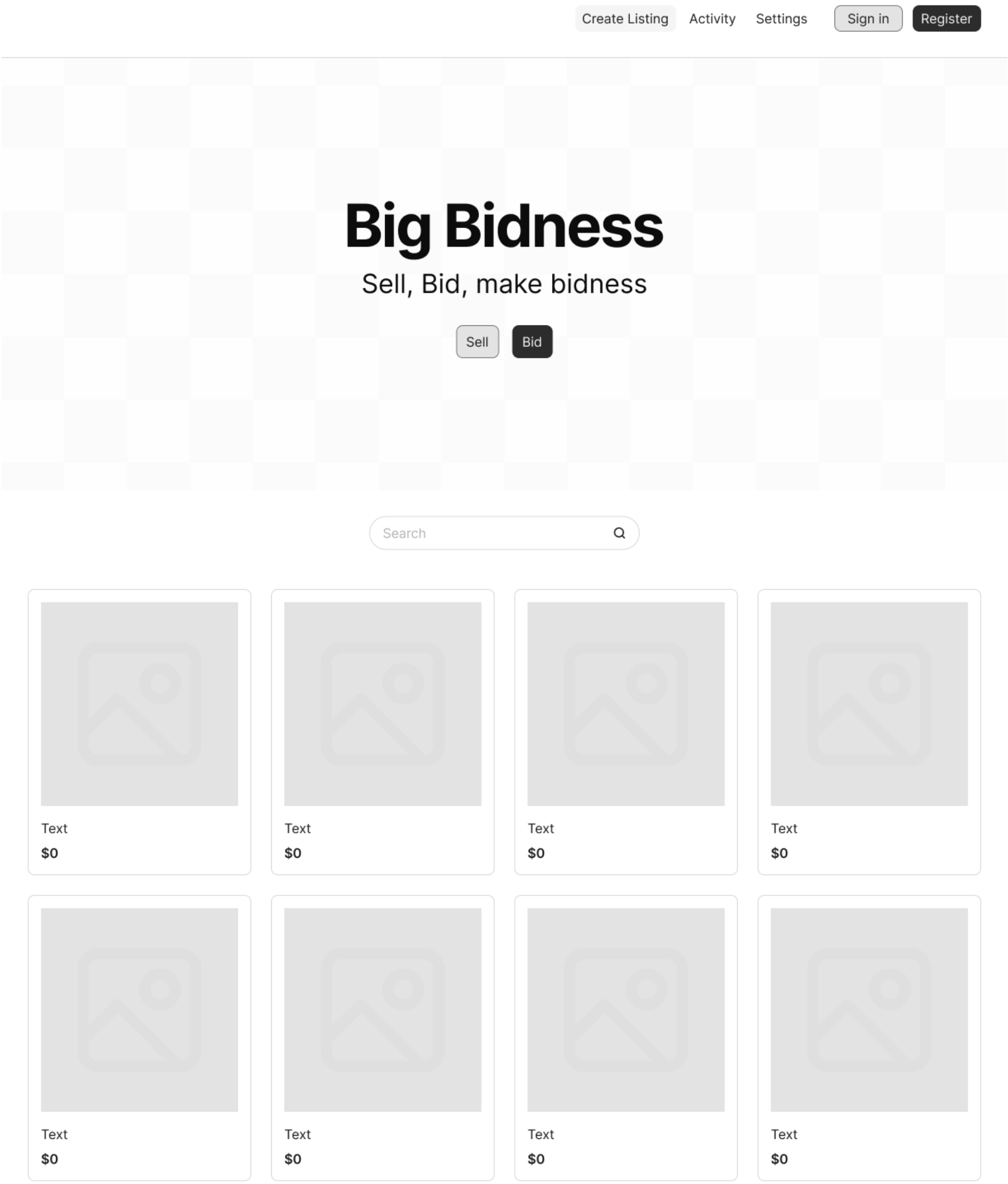
show_bids:

- Precondition: User navigates to their history page.
- Postcondition: Displays all completed bids.
- Pseudo Code:

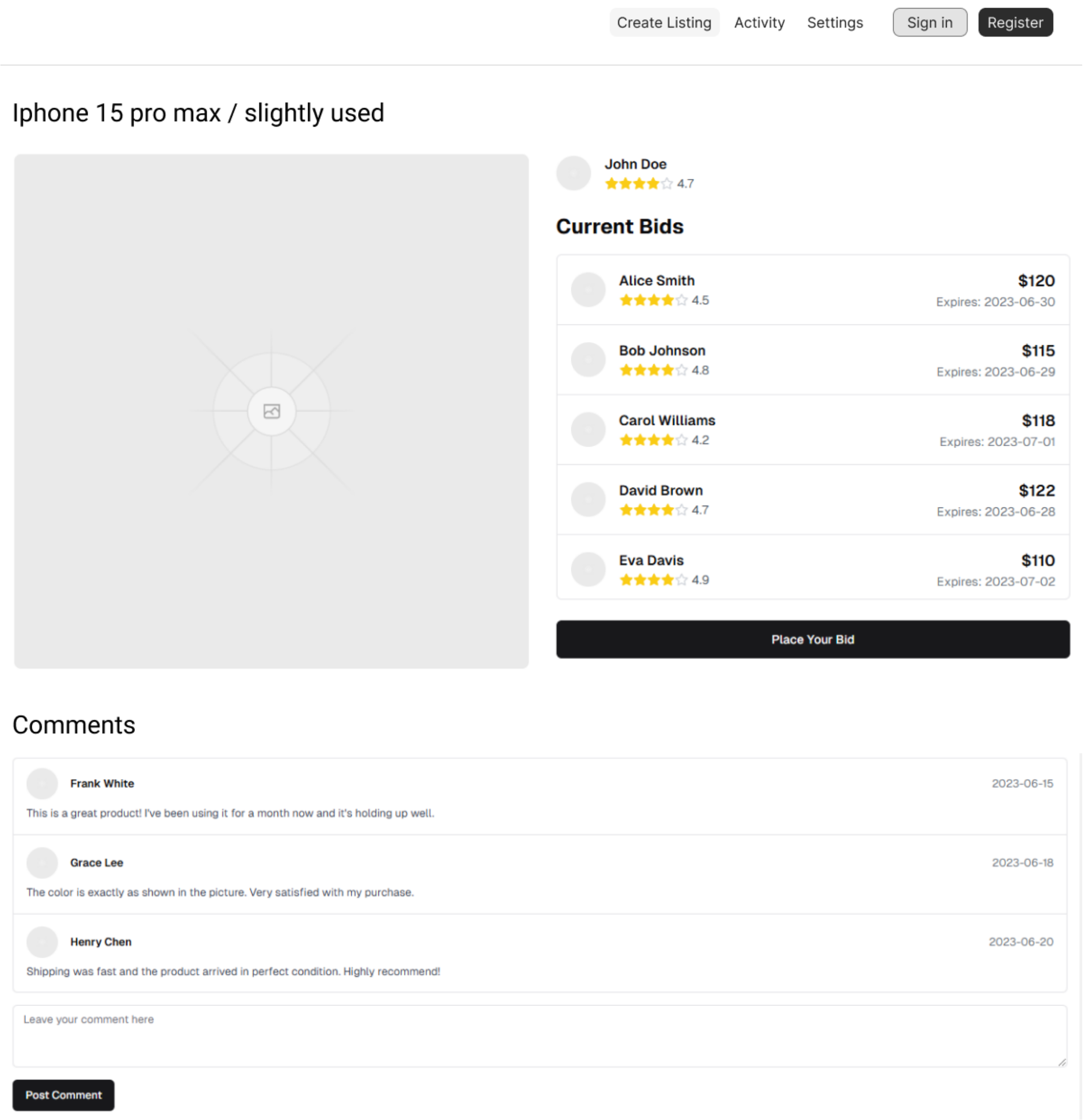
```
function viewCompletedBids(user):
            return user.getCompletedBids()
```

7 System Screens

7.1 Landing page


Create Listing
Activity
Settings
Sign in
Register
Big Bidness
Sell, Bid, make bidness
Sell
Bid
Search
Text
$0
Text
$0
Text
$0
Text
$0
Text
$0
Text
$0
Text
$0
Text
$0

7.2 Listing Detail



Iphone 15 pro max / slightly used

John Doe
★★★★☆ 4.7

Current Bids

Alice Smith	★★★★☆ 4.5	$120	Expires: 2023-06-30
Bob Johnson	★★★★☆ 4.8	$115	Expires: 2023-06-29
Carol Williams	★★★★☆ 4.2	$118	Expires: 2023-07-01
David Brown	★★★★☆ 4.7	$122	Expires: 2023-06-28
Eva Davis	★★★★☆ 4.9	$110	Expires: 2023-07-02

Comments

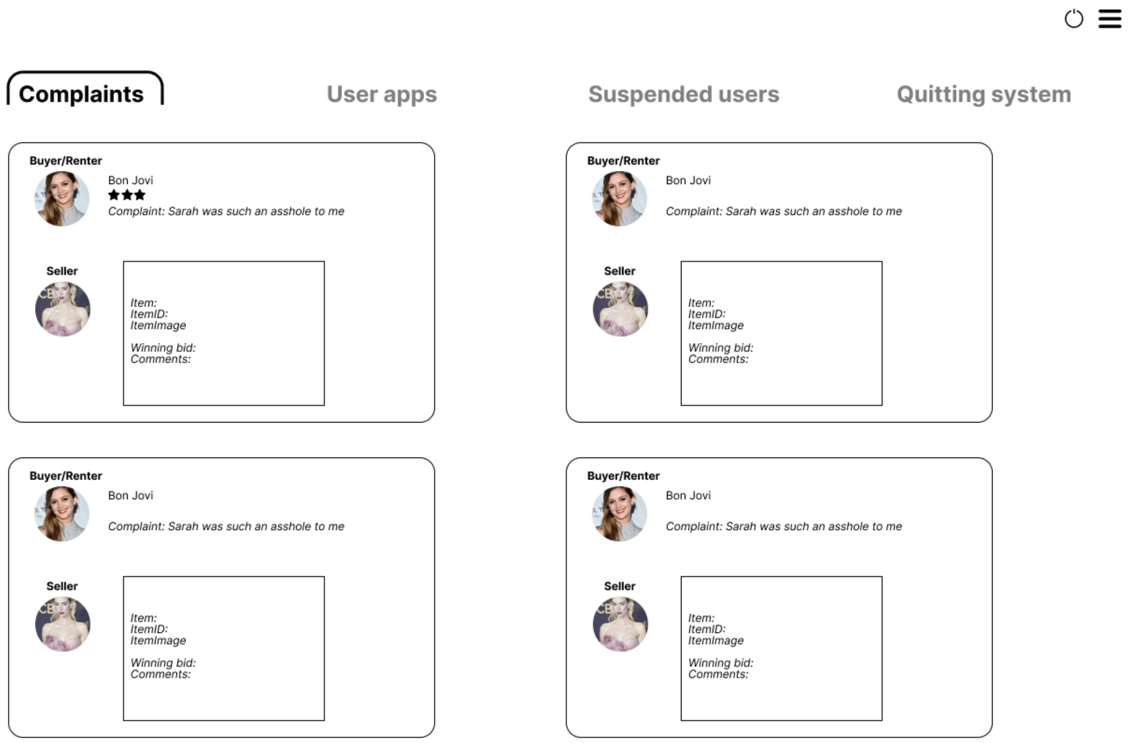
Frank White 2023-06-15
This is a great product! I've been using it for a month now and it's holding up well.

Grace Lee 2023-06-18
The color is exactly as shown in the picture. Very satisfied with my purchase.

Henry Chen 2023-06-20
Shipping was fast and the product arrived in perfect condition. Highly recommend!

Leave your comment here

7.3 Super-User Admin Portal



Buyer/Renter
Bon Jovi
★★★
Complaint: Sarah was such an asshole to me

Seller
Item:
ItemID:
ItemImage
Winning bid:
Comments:

Buyer/Renter
Bon Jovi
Complaint: Sarah was such an asshole to me

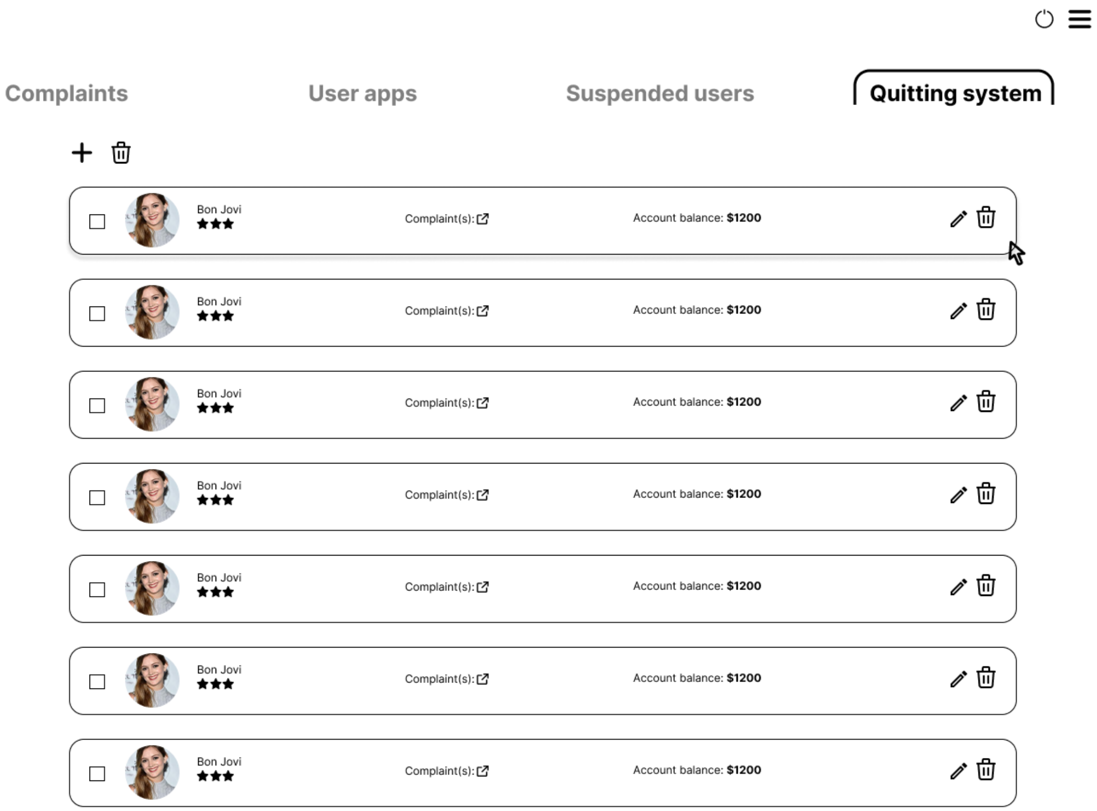
Seller
Item:
ItemID:
ItemImage
Winning bid:
Comments:

Buyer/Renter
Bon Jovi
Complaint: Sarah was such an asshole to me

Seller
Item:
ItemID:
ItemImage
Winning bid:
Comments:

Buyer/Renter
Bon Jovi
Complaint: Sarah was such an asshole to me

Seller
Item:
ItemID:
ItemImage
Winning bid:
Comments:



☐ Bon Jovi ★★★ Complaint(s): Account balance: **$1200**

☐ Bon Jovi ★★★ Complaint(s): Account balance: **$1200**

☐ Bon Jovi ★★★ Complaint(s): Account balance: **$1200**

☐ Bon Jovi ★★★ Complaint(s): Account balance: **$1200**

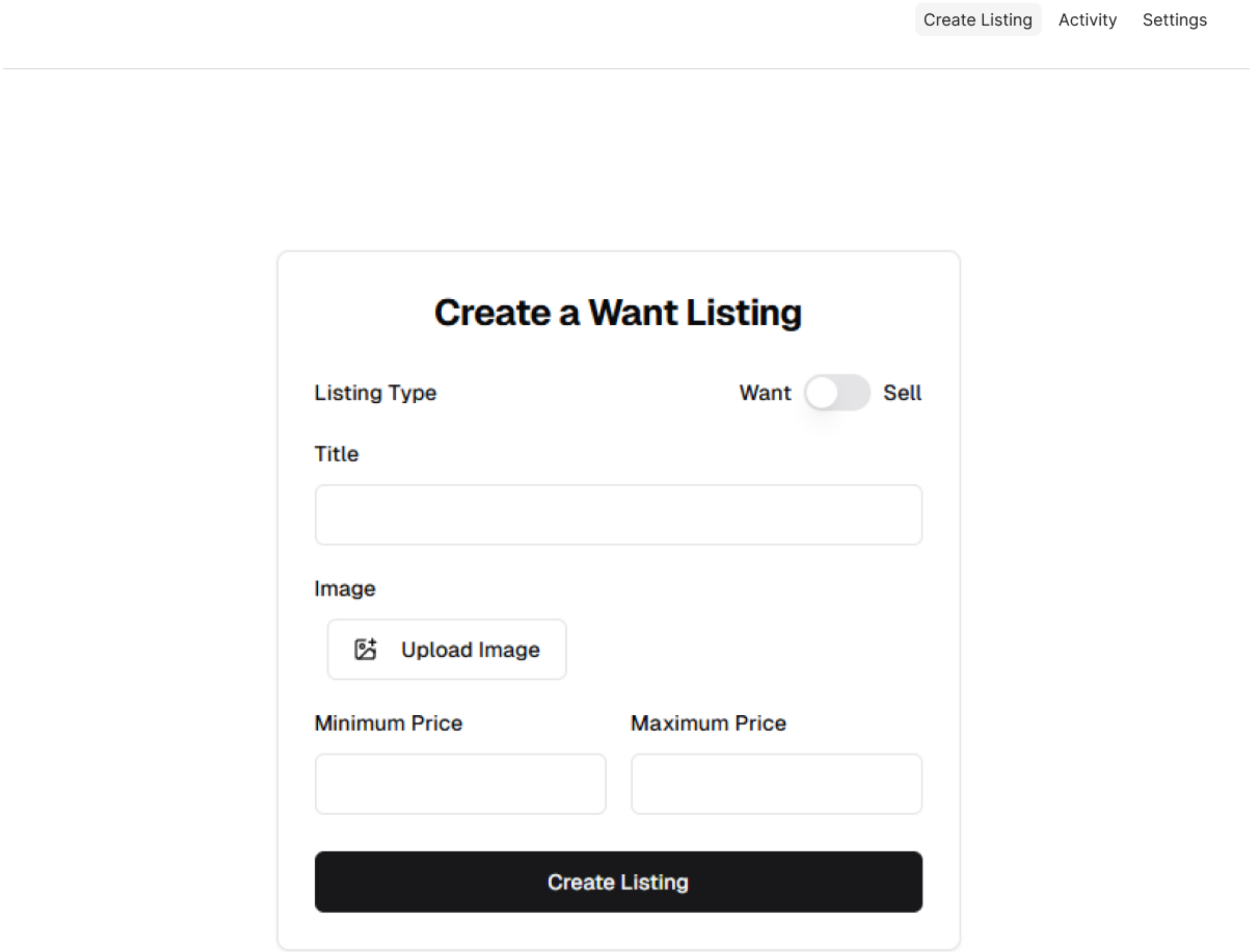
☐ Bon Jovi ★★★ Complaint(s): Account balance: **$1200**

☐ Bon Jovi ★★★ Complaint(s): Account balance: **$1200**

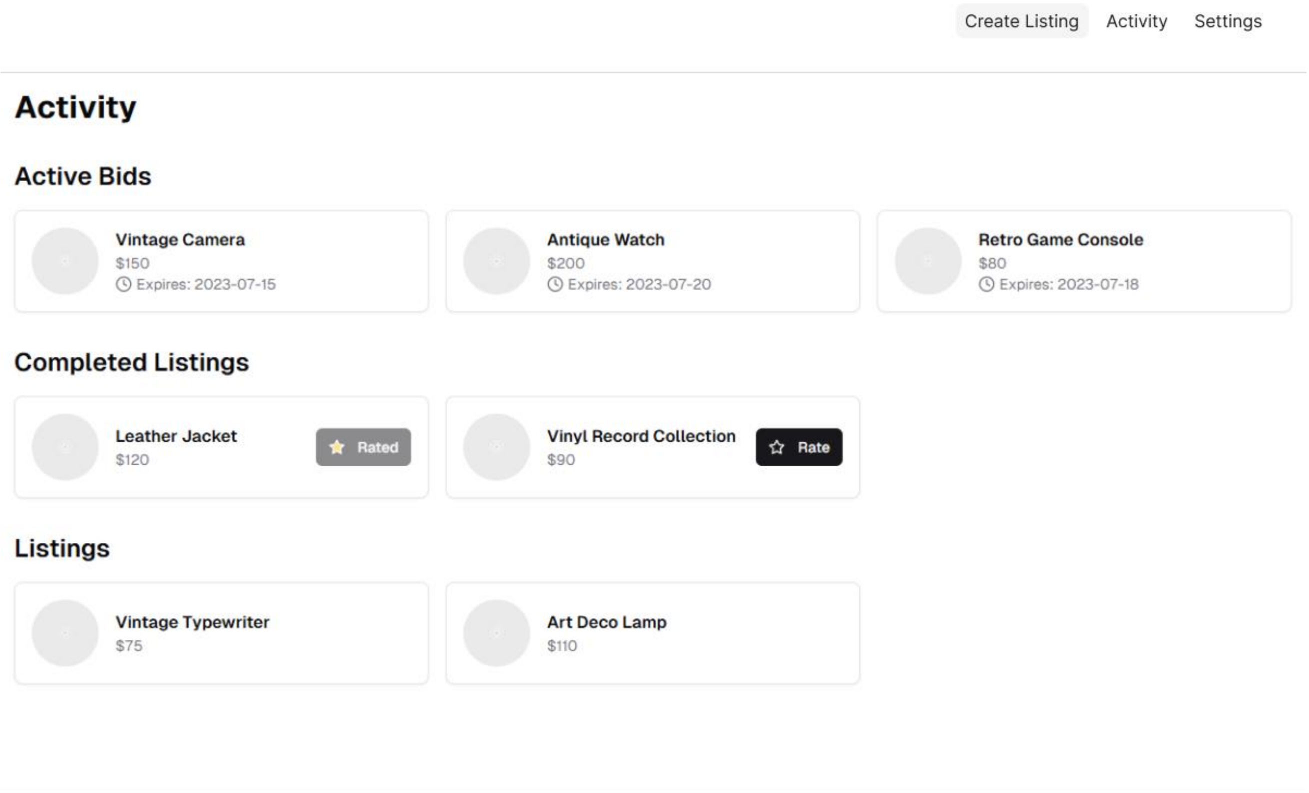
☐ Bon Jovi ★★★ Complaint(s): Account balance: **$1200**

The User apps, and Suspended users' tabs follow the same layout as the Quitting system and so to be as efficient as possible and reduce redundancy we only made the UI layout for the Quitting systems screen.

7.5 Create Listing



7.6 Activity Page



Activity

Active Bids

Vintage Camera
$150
Expires: 2023-07-15

Antique Watch
$200
Expires: 2023-07-20

Retro Game Console
$80
Expires: 2023-07-18

Completed Listings

Leather Jacket
$120
★ Rated

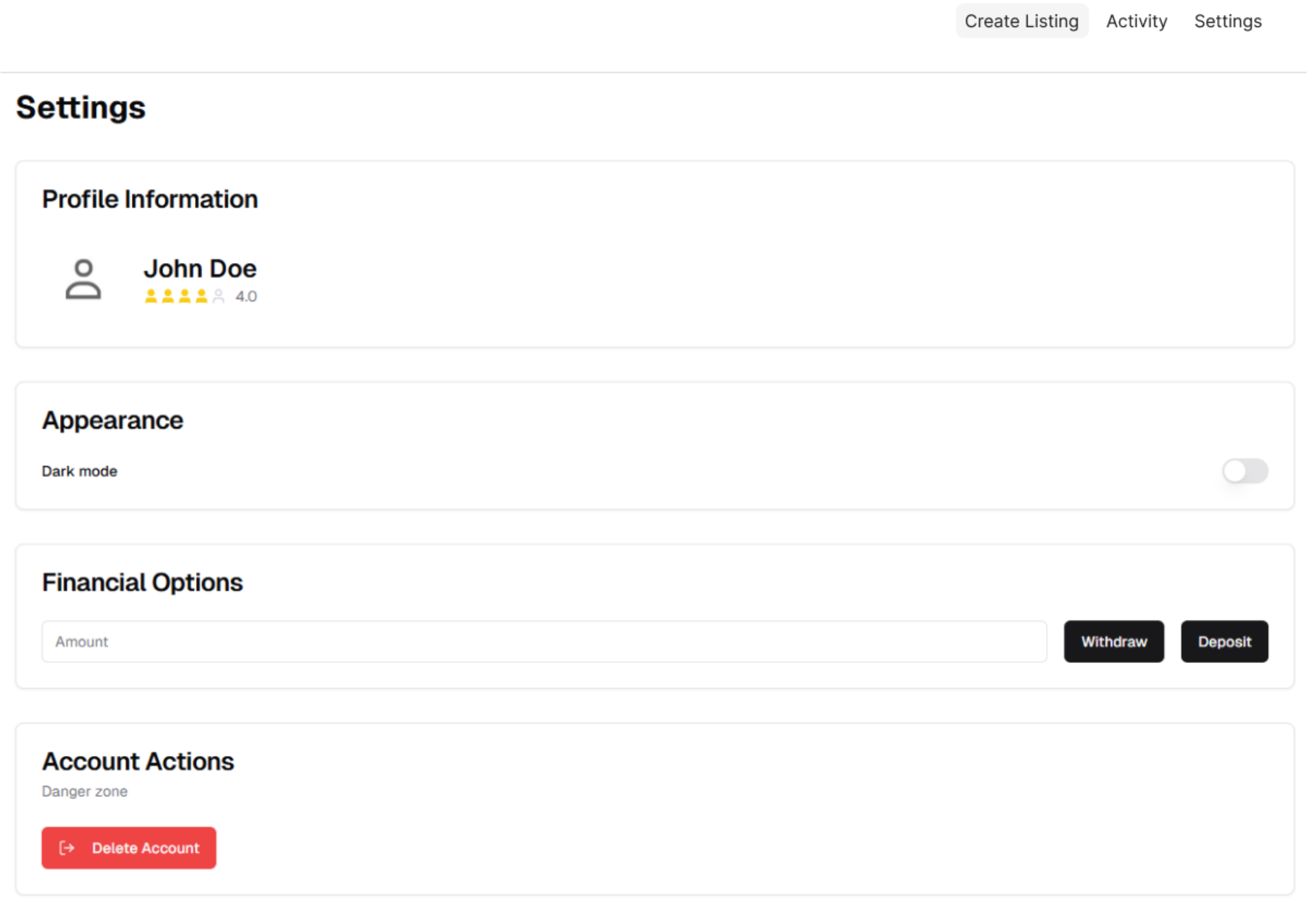
Vinyl Record Collection
$90
☆ Rate

Listings

Vintage Typewriter
$75

Art Deco Lamp
$110

7.7 Settings Page



Settings

Profile Information

John Doe
4.0

Appearance

Dark mode

Financial Options

Amount

Account Actions

Danger zone

8. GitHub Repository

Link: https://github.com/BVasquez07/big-bidness