



Online Auction App

Final Year Project Report

Submitted by

M.Umair (491-2021)

Maham Mirza (1797-2021)

Bismah Imran (1797-2021)

Supervisor

Osama Ahmed Khan

In partial fulfilment of the requirements for the degree of
Bachelor of Science in Computer Science
2025

Faculty of Engineering Sciences and Technology

Hamdard Institute of Engineering and Technology

Hamdard University, Main Campus, Karachi, Pakistan

Certificate of Approval



Faculty of Engineering Sciences and Technology

Hamdard Institute of Engineering and Technology
Hamdard University, Karachi, Pakistan

This project “_____” is presented by _____ under the supervision of their project advisor and approved by the project examination committee, and acknowledged by the Hamdard Institute of Engineering and Technology, in the fulfillment of the requirements for the Bachelor degree of _____.

Mr. XYZ
(Project Supervisor)

In-charge FYP-Committee

Mr. XYZ
(Project Co-Supervisor)

Chairman
(Department of Computing)

(Dean, FEST)

Authors' Declaration

We declare that this project report was carried out in accordance with the rules and regulations of Hamdard University. The work is original except where indicated by special references in the text and no part of the report has been submitted for any other degree. The report has not been presented to any other University for examination.

Dated:

Authors Signatures:

<Team Lead name>

<Member 2 name>

<Member 3 name>

Plagiarism Undertaking

We, M.Umair, Maham Mirza, and Bismah Imran, solemnly declare that the work presented in the Final Year Project Report titled Online Auction App has been carried out solely by ourselves with no significant help from any other person except few of those which are duly acknowledged. We confirm that no portion of our report has been plagiarized and any material used in the report from other sources is properly referenced.

Dated:

Authors Signatures:

M.Umair

Maham Mirza

Bismah Imran

Acknowledgments

Document Information

Table 1: Document Information

Customer	
Project Title	Online Auction App
Document	Final Year Project Report
Document Version	1.0
Identifier	FYP-026/FL24 Final Report
Status	Final
Author(s)	Bismah, Umair
Approver(s)	Osama Ahmed Khan
Issue Date	

Definition of Terms, Acronyms, and Abbreviations

- **Auction:** A process of selling items to the highest bidder through bidding, replicated digitally in the app.
- **Real-time Bidding:** Instant updates and bid placements, enabling a dynamic auction environment.
- **Firebase:** A cloud-based platform for app backend operations, including authentication and real-time data.
- **Flutter:** A UI toolkit used for building responsive and cross-platform applications.
- **Dart:** Programming language used with Flutter for app development.
- **Mobile Compatibility:** The ability of the app to function smoothly on smart phones and tablets.
- **Chat System:** A nin-app messaging feature enabling direct communication between buyers and sellers.
- **Encryption:** A security feature converting data into a secure format to prevent unauthorized access.
- **Notifications:** Alerts sent to users about auction updates, bids, or messages.
- **Admin Module:** A backend feature for administrators to manage auctions, users, and disputes.

Abstract

The Online Auction App is a modern platform that makes buying and selling items more engaging and flexible. Unlike traditional shopping websites with fixed prices, this app allows users to bid for items in real-time, ensuring competitive deals for buyers and better profits for sellers. Built with mobile users in mind, the app provides a user-friendly experience with features like instant notifications, secure logins, and a chat system for direct communication. Backed by reliable technologies like Firebase and Flutter, it ensures smooth and secure operations. Whether you're looking to buy unique items or sell to a wider audience, the app offers an exciting and interactive auction environment.

Table of Contents

Certificate of Approval	2
Authors' Declaration	3
Acknowledgments	5
Document Information	6
Abstract	7
List of Figures	12
List of Tables	13
CHAPTER 1	14
INTRODUCTION	14
1.1 Motivation	14
1.2 Problem Statement	14
1.3 Goals and Objectives	14
1.4 Project Scope	15
CHAPTER 2	16
RELEVANT BACKGROUND & DEFINITIONS	16
CHAPTER 3	20
LITERATURE REVIEW & RELATED WORK	20
Literature Review	20
Related Work	20
Gap Analysis	20
CHAPTER 4	22
1. Software Engineering Methodology	22
2. Project Methodology	22
3. Phases of Project	22
4. Software/Tools that Used in Project	23
5. Hardware that Used in Project	23
Chapter 5	24
4.1 Proposed System Architecture/Design	24
4.2 Functional Specifications	24
4.3 Non-Functional Specifications	24
4.4 Testing	25
4.5 Purpose of Testing	25
4.6 Test Cases	25
Chapter 5	Error! Bookmark not defined.
EXPERIMENTAL EVALUATIONS & RESULTS	27
Evaluation Testbed	27

Results and Discussion	27
CHAPTER 6	29
CONCLUSION AND DISCUSSION	29
7.1 Strength of this Project	29
7.2 Limitations and Future Work	29
7.3 Reasons for Failure – If Any	29
REFERENCES	29
APPENDICES	31
A0. COPY OF PROJECT REGISTRATION FORM	32
A1A. PROJECT PROPOSAL AND VISION DOCUMENT	33
A1B. COPY OF PROPOSAL EVALUATION COMMENTS BY JURY	Error! Bookmark not defined.
A2. REQUIREMENT SPECIFICATIONS	2
A3. DESIGN SPECIFICATIONS	19
A4. OTHER TECHNICAL DETAIL DOCUMENTS	59
Test Cases Document	59
UI/UX Detail Document	66
Coding Standards Document	68
Project Policy Document	70
User Manual Document	72
A5. FLYER & POSTER DESIGN	Error! Bookmark not defined.
A6. COPY OF EVALUATION COMMENTS	Error! Bookmark not defined.
COPY OF EVALUATION COMMENTS BY SUPERVISOR FOR PROJECT – I MID SEMESTER EVALUATION	Error! Bookmark not defined.
COPY OF EVALUATION COMMENTS BY JURY FOR PROJECT – I END SEMESTER EVALUATION	Error! Bookmark not defined.
COPY OF EVALUATION COMMENTS BY SUPERVISOR FOR PROJECT – II MID SEMESTER EVALUATION	Error! Bookmark not defined.
A7. MEETINGS' MINUTES & Sign-Off Sheet	Error! Bookmark not defined.
A8. DOCUMENT CHANGE RECORD	Error! Bookmark not defined.
A9. PROJECT PROGRESS	Error! Bookmark not defined.
A10. RESEARCH PAPER	Error! Bookmark not defined.
A11. Plagiarism Test Summary Report	74

CHAPTER 1

INTRODUCTION

The Online Auction App is a cutting-edge platform designed to make buying and selling items easier and more exciting. Unlike traditional e-commerce websites where items have fixed prices, this app allows users to participate in live auctions, giving buyers the chance to snag great deals and sellers the opportunity to get the best possible price for their items. Whether you're looking to

auction off collectibles, electronics, or even vehicles, this app offers a seamless and fun way to connect with potential buyers and sellers from anywhere. One of the key features of the app is its real-time bidding system. Users can see the current bids and place their own offers in a matter of seconds. This creates a dynamic marketplace where competition drives better prices for everyone. To keep things secure and organized, the app also has user profiles, where buyers and sellers can manage their information, track their bidding history, and communicate directly through a built-in chat system. Notifications keep users updated on the status of their auctions, ensuring they never miss out on important updates. Designed for mobile devices, the app is accessible and easy to use, even for those who are always on the go. It is built using modern technology to ensure fast and reliable performance, with a focus on keeping user data safe and secure. Whether you're a seasoned bidder or just getting started, the Online Auction App provides a simple, secure, and engaging way to explore the world of online auctions.

1.1 Motivation

The motivation behind developing the Online Auction App stems from the need to create a more interactive and flexible marketplace for buying and selling goods. Traditional ecommerce platforms often lack the dynamic nature of auctions, where buyers can bid for items and potentially get them at a lower price, while sellers can benefit from competitive bidding. The app aims to bridge this gap by providing a platform that offers real-time bidding, enhancing the shopping experience for both buyers and sellers. It encourages a fair marketplace where users can engage more actively and enjoy the thrill of auctions from the comfort of their mobile devices.

1.2 Problem Statement

Current e-commerce applications typically operate on fixed pricing models, which do not allow for price negotiation or bidding. This static pricing approach limits both buyers and sellers, as buyers cannot compete for better deals, and sellers cannot maximize their profits through competitive bidding. The absence of a real-time auction system in most e-commerce apps creates a gap in the market, leaving users seeking more dynamic and flexible options.

The Online Auction App addresses this issue by introducing a platform that facilitates real time bidding, thereby creating a more engaging and profitable environment for all users.

1.3 Goals and Objectives

The primary goal of the Online Auction App is to provide a secure, user-friendly platform for online auctions, where users can buy or sell goods dynamically through bidding. The objectives include:

- Enhancing the buying experience by offering competitive deals through real time auctions.
- Empowering sellers to maximize their profits by reaching a broader audience and receiving competitive bids.
- Facilitating real time interactions between buyers and sellers for negotiations and updates.
- Ensuring the app is optimized for mobile devices, providing a seamless user experience on the go.

1.3 Project Scope

- The project scope encompasses several key features and functionalities to ensure a comprehensive online auction platform:
- User Management: Users can register, log in, and manage their profiles securely.
- Auction Listings: Users can list items for auction, including details like start and end times, and minimum bids.
- Bidding: The app supports real-time bidding, allowing users to place and track bids dynamically.
- Chat System: Buyers and sellers can communicate through a chat system to negotiate and discuss terms.

- Administration: An admin module to manage user accounts, auctions, and resolve disputes.
- Notifications: Real-time notifications for updates on bids and auction statuses.
- Mobile Compatibility: The app is designed to be responsive and accessible on all mobile devices, ensuring a broad user base and ease of use.

CHAPTER 2

RELEVANT BACKGROUND & DEFINITIONS

Relevant Background & Definitions The Online Auction App represents a significant advancement in the way goods and services are bought and sold in the digital marketplace. To fully understand its importance and functionality, it is essential to delve into the background of online auctions, the technological infrastructure supporting the app, and the definitions of key terms and concepts related to this domain.

Historical Context of Online Auctions Online auctions have evolved from traditional auction houses, where bidders gather in person to compete for items, to digital platforms that allow participants to engage in the auction process from anywhere in the world. The first significant foray into online auctions began in the mid-1990s with platforms like eBay, which revolutionized the way consumers could buy and sell items. These platforms introduced the concept of bidding on items over the internet, opening up new opportunities for both buyers and sellers. The convenience of participating in auctions from the comfort of one's home, coupled with the potential for finding rare items or getting good deals, led to the rapid growth of online auction platforms. As technology advanced, so did the features of online auction systems. Today, these platforms support real-time bidding, mobile compatibility, and secure payment systems, making them more accessible and user-friendly than ever before. The development of the Online Auction App builds on this legacy, incorporating modern technological advancements to provide a seamless, interactive, and secure auction experience.

Technological Infrastructure The Online Auction App is built using Flutter and Dart, a combination that allows for the development of highly responsive, cross-platform mobile applications. Flutter, an open-source UI software development kit by Google, enables the creation of natively compiled applications for mobile, web, and desktop from a single codebase. It provides a rich set of pre-designed widgets and tools, making it easier for developers to build aesthetically pleasing and functional user interfaces. Dart, the programming language used with Flutter, is designed for client development, such as for web and mobile apps. It is optimized for fast development, offering a smooth and quick development cycle, which is crucial for dynamic applications like the Online Auction App. The use of Flutter and Dart ensures that the app is not only visually appealing but also efficient and fast. The backend of the app is powered by Google Firebase, a comprehensive platform for app development. Firebase provides various services such as real-time database management

List of Figures

Figure No	Description	Page No.
FIGURE 1. 1 : SCOPE OF PROJECT		2
FIGURE 2. 1 : FACE DETECTION		4
FIGURE 2. 2 : FACE RECOGNITION PROCESS		5
FIGURE 4. 1 : SYSTEM FLOW FOR FACE RECOGNITION		9
FIGURE 4. 2 : USE CASE MODEL		10
FIGURE 4. 3 : USE CASE DIAGRAM FOR SETTING		12
FIGURE 5. 1 : DESIGN OF A SYSTEM		13
FIGURE 6. 1 : STEP 1 DOWNLOADS OPENCV MANAGER		18
FIGURE 6. 2 : STEP 2 INSTALLS APPLICATION		19
FIGURE 6. 3 : STEP 3 OPEN APPLICATION		20
FIGURE 6. 4 : STEP 4 OPEN INBOX		21
FIGURE 6. 5 : STEP 5 SELECT MESSAGE		22
FIGURE 6. 6 : STEP 6 TEXT VARIATIONS		23
FIGURE 6. 7 : STEP 7 READ CONTACTS		24
FIGURE 6. 8 : STEP 8 FOR WRITE MESSAGE		25
FIGURE 6.10 : STEP 10 SETTINGS		26

List of Tables

Table No.	Description	Page No.
TABLE 2.1: COMPARISON TABLE		6
TABLE 3.1: PHASES OF PROJECT		7
TABLE 5.1: TEST CASE 1		16
TABLE 5.2: TEST CASE 2		16
TABLE 5.3: TEST CASE 3		16
TABLE 5.4: TEST CASE 4		17
TABLE 5.5: TEST CASE 5		17
TABLE 5.6: TEST CASE 6		17

CHAPTER 1

INTRODUCTION

The Online Auction App is a cutting-edge platform designed to make buying and selling items easier and more exciting. Unlike traditional e-commerce websites where items have fixed prices, this app allows users to participate in live auctions, giving buyers the chance to snag great deals and sellers the opportunity to get the best possible price for their items. Whether you're looking to auction off collectibles, electronics, or even vehicles, this app offers a seamless and fun way to connect with potential buyers and sellers from anywhere. One of the key features of the app is its real-time bidding system. Users can see the current bids and place their own offers in a matter of seconds. This creates a dynamic marketplace where competition drives better prices for everyone. To keep things secure and organized, the app also has user profiles, where buyers and sellers can manage their information, track their bidding history, and communicate directly through a built-in chat system. Notifications keep users updated on the status of their auctions, ensuring they never miss out on important updates. Designed for mobile devices, the app is accessible and easy to use, even for those who are always on the go. It is built using modern technology to ensure fast and reliable performance, with a focus on keeping user data safe and secure. Whether you're a seasoned bidder or just getting started, the Online Auction App provides a simple, secure, and engaging way to explore the world of online auctions.

1.1 Motivation

The motivation behind developing the Online Auction App stems from the need to create a more interactive and flexible marketplace for buying and selling goods. Traditional ecommerce platforms often lack the dynamic nature of auctions, where buyers can bid for items and potentially get them at a lower price, while sellers can benefit from competitive bidding. The app aims to bridge this gap by providing a platform that offers real time bidding, enhancing the shopping experience for both buyers and sellers. It encourages a fair marketplace where users can engage more actively and enjoy the thrill of auctions from the comfort of their mobile devices.

1.2 Problem Statement

Current e-commerce applications typically operate on fixed pricing models, which do not allow for price negotiation or bidding. This static pricing approach limits both buyers and sellers, as buyers cannot compete for better deals, and sellers cannot maximize their profits through competitive bidding. The absence of a real time auction system in most e-commerce apps creates a gap in the market, leaving users seeking more dynamic and flexible options. The Online Auction App addresses this issue by introducing a platform that facilitates real time bidding, thereby creating a more engaging and profitable environment for all users.

1.3 Goals and Objectives

The primary goal of the Online Auction App is to provide a secure, user friendly platform for online auctions, where users can buy or sell goods dynamically through bidding. The objectives include:

- Enhancing the buying experience by offering competitive deals through real time auctions.
- Empowering sellers to maximize their profits by reaching a broader audience and receiving competitive bids.
- Facilitating real-time interactions between buyers and sellers for negotiations and updates.
- Ensuring the app is optimized for mobile devices, providing a seamless user experience on the go.

1.4 Project Scope

The project scope encompasses several key features and functionalities to ensure a comprehensive online auction platform:

- **User Management:** Users can register, log in, and manage their profiles securely.
- **Auction Listings:** Users can list items for auction, including details like start and end times, and minimum bids.
- **Bidding:** The app supports real-time bidding, allowing users to place and track bids dynamically.
- **Chat System:** Buyers and sellers can communicate through a chat system to negotiate and discuss terms.
- **Administration:** An admin module to manage user accounts, auctions, and resolve disputes.
- **Notifications:** Real time notifications for updates on bids and auction statuses.
- **Mobile Compatibility:** The app is designed to be responsive and accessible on all mobile devices, ensuring a broad user base and ease of use.

CHAPTER 2

RELEVANT BACKGROUND & DEFINITIONS

The Online Auction App represents a significant advancement in the way goods and services are bought and sold in the digital marketplace. To fully understand its importance and functionality, it is essential to delve into the background of online auctions, the technological infrastructure supporting the app, and the definitions of key terms and concepts related to this domain.

Historical Context of Online Auctions

Online auctions have evolved from traditional auction houses, where bidders gather in person to compete for items, to digital platforms that allow participants to engage in the auction process from anywhere in the world. The first significant foray into online auctions began in the mid-1990s with platforms like eBay, which revolutionized the way consumers could buy and sell items. These platforms introduced the concept of bidding on items over the internet, opening up new opportunities for both buyers and sellers. The convenience of participating in auctions from the comfort of one's home, coupled with the potential for finding rare items or getting good deals, led to the rapid growth of online auction platforms. As technology advanced, so did the features of online auction systems. Today, these platforms support real-time bidding, mobile compatibility, and secure payment systems, making them more accessible and user-friendly than ever before. The development of the Online Auction App builds on this legacy, incorporating modern technological advancements to provide a seamless, interactive, and secure auction experience.

Technological Infrastructure

The Online Auction App is built using Flutter and Dart, a combination that allows for the development of highly responsive, cross-platform mobile applications. Flutter, an open source UI software development kit by Google, enables the creation of natively compiled applications for mobile, web, and desktop from a single codebase. It provides a rich set of pre-designed widgets and tools, making it easier for developers to build aesthetically pleasing and functional user interfaces. Dart, the programming language used with Flutter, is designed for client development, such as for web and mobile apps. It is optimized for fast development, offering a smooth and quick development cycle, which is crucial for dynamic applications like the Online Auction App. The use of Flutter and Dart ensures that the app is not only visually appealing but also efficient and fast. The backend of the app is powered by Google Firebase, a comprehensive platform for app development. Firebase provides various services such as real-time database management, user authentication, and cloud messaging. These features are integral to the functionality of the Online

Auction App, allowing it to handle real-time data updates, secure user logins, and instant notifications effectively.

Definitions and Key Concepts

To fully grasp the workings and benefits of the Online Auction App, it is important to understand several key concepts and terms associated with online auctions and the app's features:

1. **Auction:** A public sale in which goods or property are sold to the highest bidder. Online auctions replicate this process digitally, allowing users to bid on items through an internet platform.
2. **Real-time Bidding:** A process in which auction bids are placed, updated, and displayed instantaneously. This feature is crucial for online auctions, as it provides a dynamic and engaging user experience.
3. **User Authentication:** A security process that ensures only authorized users can access certain areas of the app or perform specific actions. The Online Auction App uses Firebase Authentication to manage user sign-ins, sign-ups, and email verification.
4. **Firebase Real-time Database:** A cloud-hosted No SQL database that allows data to be stored and synchronized in real-time across all connected clients. This is essential for keeping auction listings and bid updates current and consistent for all users.
5. **Mobile Compatibility:** The ability of an app to function seamlessly across various mobile devices, ensuring that users can access and use the app from their smart phones or tablets without any issues.
6. **Chat System:** A built-in messaging platform that enables users to communicate directly with each other. In the context of the Online Auction App, this allows buyers and sellers to discuss items, negotiate prices, and finalize transactions more effectively.
7. **Admin Module:** A part of the app designed for administrators to manage auctions, oversee user accounts, and resolve disputes. This ensures the smooth operation of the platform and maintains the integrity of the auction process.
8. **Notifications:** Alerts sent to users to inform them of important updates, such as being outbid on an item or an auction ending soon. Notifications are crucial for keeping users engaged and informed about their activities on the platform.

9. Encryption: The process of converting data into a secure format that cannot be read by unauthorized users. This is an important feature for protecting sensitive user information and ensuring the security of transactions.

10. User Interface : The part of the app that users interact with. A well-designed UI is crucial for providing a positive user experience, making it easy for users to navigate the app and participate in auctions.

The Role of Online Auction Apps in Modern Commerce

Online auction apps like the one described in this document play a pivotal role in modern commerce by offering a flexible and dynamic alternative to traditional e-commerce platforms. They cater to a variety of users, including individual sellers looking to reach a

broader audience, collectors seeking rare items, and businesses aiming to maximize their revenue streams. These apps benefit users by:

- Providing Competitive Prices: Buyers can potentially get items at lower prices through competitive bidding, while sellers can achieve higher profits by attracting multiple bidders.
- Expanding Market Reach: Sellers can reach a global audience, increasing the chances of finding the right buyer for their items.
- Enhancing User Engagement: The interactive nature of auctions, with real-time updates and competitive bidding, creates a more engaging shopping experience compared to fixed-price models.

Challenges and Considerations

While online auction apps offer numerous benefits, they also present certain challenges that need to be addressed:

- Security Concerns: Protecting user data and ensuring secure transactions is paramount. Implementing robust authentication methods and data encryption is essential to gaining user trust.
- User Experience: The app must be easy to use and navigate, even for those who may not be tech-savvy. A cluttered or confusing interface can deter users from participating in auctions.
- Real-time Performance: The app must handle real-time data efficiently to provide a seamless experience. Any delays or lags can frustrate users and impact the overall success

of the auctions.

CHAPTER 3

LITERATURE REVIEW & RELATED WORK

Literature Review

Online auctions have been a popular method of buying and selling goods for years, revolutionized by platforms like eBay and similar websites. These platforms introduced the concept of bidding on items online, allowing users to participate in auctions without being physically present. Over time, this model has evolved with advancements in technology, leading to the development of more sophisticated auction systems that are faster, more secure, and accessible through mobile devices. Several studies and developments have focused on enhancing online auction platforms. Research has been conducted on improving user experience, ensuring data security, and optimizing real-time bidding processes. Various papers have highlighted the importance of user engagement and the role of instant notifications in keeping users informed and involved. There has also been significant work on using mobile technology to make auction platforms more accessible, catering to the growing number of users who prefer conducting transactions via their smart phones

Related Work

1. eBay and Other Online Platforms:

eBay was one of the first major online auction platforms, setting the standard for online auctions. It allowed users to bid on a wide range of items, from collectibles to electronics, and introduced features like auction timers, feedback systems, and secure payment methods. Other platforms, such as Amazon Auctions and Yahoo! Auctions, followed suit, offering similar services with varying degrees of success.

2. Mobile Auction Apps:

With the rise of mobile technology, several auction platforms developed apps to cater to mobile users. These apps focused on providing a seamless experience on smaller screens, with features like swipe navigation, push notifications, and mobile payment options. Research in this area has shown that mobile compatibility significantly increases user engagement and bidding activity.

3. Security Enhancements:

Ensuring the security of user data and transactions has been a critical area of focus. Studies have explored the use of encryption, secure login systems, and fraud detection algorithms to protect users and maintain the integrity of the auction process. Platforms that fail to implement robust security measures often face user distrust and potential legal issues.

4. User Engagement and Gamification:

Some platforms have experimented with gamification elements to enhance user engagement. Features like reward points for participation, leaderboards, and interactive tutorials have been shown to increase user activity and retention. Research indicates that creating a fun and engaging environment can lead to higher bidding rates and user satisfaction.

Gap Analysis

Despite the progress made in online auction platforms, there are still several gaps and areas for improvement:

1. Lack of Real-time Interaction: Many traditional auction platforms, while effective, lack real-time interaction capabilities. This gap is particularly noticeable in mobile apps, where users expect

instant feedback and updates. The Online Auction App addresses this gap by integrating real-time bidding and notifications, ensuring users are always up-to-date with the latest auction status.

2. Limited Mobile Optimization: While many platforms offer mobile apps, not all are fully optimized for mobile use. Issues like slow loading times, difficult navigation, and limited functionality can deter users from participating in auctions on their mobile devices. The Online Auction App is designed with mobile-first principles, ensuring a smooth and responsive experience on all mobile devices.

3. Security Concerns: Despite advances in security technology, some platforms still struggle with protecting user data and ensuring safe transactions. The Online Auction App fills this gap by using secure authentication methods and encryption to safeguard user information, providing a secure environment for both buyers and sellers.

4. User Experience and Accessibility: Many existing platforms can be overwhelming or confusing for new users, with complex interfaces and jargon filled instructions. This limits the platform's reach and user base. The Online Auction App focuses on providing a user friendly interface with clear instructions and easy navigation, making it accessible to users of all technical abilities.

5. Limited Communication Features: Communication between buyers and sellers is often limited to post-sale interactions or external messaging platforms. This can lead to misunderstandings and disputes. The Online Auction App includes an integrated chat system, allowing users to communicate directly within the app, fostering transparency and reducing the likelihood of disputes.

CHAPTER 4

PROJECT DISCUSSION

1. Software Engineering Methodology

To develop the Online Auction App, we adopted the Agile Software Development Methodology. This methodology is known for its flexibility, iterative development, and continuous delivery approach. Agile was chosen because it fits perfectly with the requirements of a modern mobile application where user feedback and quick updates are important.

In Agile, the work is broken down into small time-bound cycles known as sprints. Each sprint focuses on developing specific features and includes planning, design, development, testing, and feedback. This allowed us to gradually build and improve the app while ensuring high-quality results. Regular meetings and reviews were conducted to track progress and make changes where necessary.

2. Project Methodology

Our project methodology follows a step-by-step and user-focused approach, structured as follows:

1. Requirement Analysis: Understanding what features users need, such as bidding, chat, and secure login.
2. Planning: Dividing the project into phases and assigning tasks to team members.
3. Design: Creating wireframes and UI designs using Flutter to ensure a clean and responsive interface.
4. Development: Writing code for frontend and backend, integrating features like Firebase authentication, real-time bidding, and push notifications.
5. Testing: Checking each module for bugs and performance issues after every sprint.
6. Deployment: Finalizing the app and preparing it for release on mobile platforms.
7. Maintenance: Making improvements based on user feedback and fixing post deployment issues.

3. Phases of Project

The project was divided into the following major phases:

3.1 Planning and Research :

Identified the project goals and main features. Researched existing auction platforms to gather ideas and inspiration.

3.2 Design :

Created UI wireframes and defined user flow. Designed the overall layout and experience of the app using Flutter.

3.3 Development :

Set up Firebase for authentication and real-time database. Developed features like login/signup, bidding system, notifications, and chat.

3.4 Testing :

Conducted manual testing for each feature. Fixed bugs and improved app speed and user

experience.

3.5 Deployment :

Finalized and published the application. Prepared project documentation and presentation materials.

4. Software/Tools that Used in Project

Below are the main software and tools used during development:

- **Flutter:** Used to develop the mobile app with cross platform support.
- **Firebase:** Backend services including authentication, real-time database, and cloud storage.
- **Visual Studio Code :** Code editor used for writing and managing the app's codebase.
- **Figma:** For UI/UX design and wire framing.
- **Git Hub:** Version control and project collaboration.
- **Android Studio:** Emulator and testing environment for mobile app

5. Hardware that Used in Project

The development of the project required basic and widely available hardware:

- **Purpose Laptop/PC:** Core i5 or higher, 8GB RAM used for development and testing
- **Smartphone:** Android device for real-time app testing
- **Internet Connection:** Required for Firebase services, Git Hub, and collaboration.
- **Storage Devices:** USB drives and hard drives for backup and data transfer

Chapter 5

IMPLEMENTATION

5.1 Proposed System Architecture/Design

The Online Auction App is designed with a client-server architecture using Flutter for the frontend and Firebase for the backend. The architecture ensures real time communication, secure authentication, and seamless user experience across devices.

5.1.1 Client Side:

- Built with Flutter to support Android platforms.
- Handles user interactions like bidding, login, chat, and viewing auctions.
- Communicates with Firebase for data storage and updates.

5.1.2 Backend:

Firebase Authentication: Manages secure user login and registration.

- Cloud Fire store: Stores auction item data, user profiles, and bids in real-time.
- Firebase Cloud Messaging : Sends instant notifications for bid updates.
- Firebase Storage: Stores item images and user uploads securely.

5.1.3 System Flow

- User logs in or registers via the app.
- Firebase verifies credentials and provides access.
- Users can list items, place bids, or communicate via chat.
- Updates are reflected instantly using real-time database features

5.2 Functional Specifications

These are the features and operations the system is expected to perform:

- User Authentication: Sign up, login, and logout functionality.
- Item Listing: Sellers can post products for auction with images, description, and starting bid.
- Bidding System: Buyers can place bids on items in real time.
- Notifications: Instant alerts for bid updates and auction results.
- Chat System: Direct communication between buyer and seller within the app.
- Profile Management: Users can manage personal information and view past auctions or bids.

5.3 Non-Functional Specifications

These refer to the quality attributes of the system:

- Performance: The app provides fast response times with minimal delays.
- Scalability: Firebase supports multiple concurrent users without performance degradation.
- Security: User data and authentication are protected using Firebase Authentication and secure storage.
- Usability: The app has a clean, intuitive user interface designed for ease of use on mobile devices.
- Reliability: Real time updates and consistent data synchronization ensure smooth operation.
- Portability: The app runs on both Android and iOS platforms using Flutter.

5.4 Testing

The application was tested thoroughly to ensure all features work as intended and that there are no major bugs or issues.

Manual testing was conducted during and after each development sprint.

5.4.1 Types of Testing Used

- Unit Testing: Individual components like login, bidding, and item listing were tested separately.
- Integration Testing: Combined modules such as bidding with notifications and item display with chat were tested together.
- UI Testing: Ensured that the user interface works smoothly on different screen sizes and devices.
- Real Device Testing: The app was tested on actual Android smart phones to evaluate performance and usability.

5.5 Purpose of Testing

The main objectives of testing the Online Auction App were:

- To verify that all features work correctly.
- To identify and fix bugs during development.
- To ensure the app is user-friendly and provides a smooth experience.
- To make sure the app meets all functional and non functional requirements.
- To check the real time performance of features like bidding and messaging.

5.6 Test Cases

Test Case	Description	Expected Results	Actual Results	Status
Login with valid credentials	User enters correct email & password	User logs in successfully	Works as expected	Pass
Login with wrong password	User enters incorrect password	Shows error message	Shows error message	Pass
Add new auction item	Seller fills form and uploads image	Item is listed successfully	Works correctly	Pass
Place a bid	User places a higher bid than current	Bid is accepted and updated	Bid updates in real time	Pass
Notification in new bid	New bid is placed on item	Notify seller and other bidders	Notifications received	Pass
Start chat with seller	Buyer clicks on chat button	Chat opens with correct user	Chat functions properly	Pass

Chapter 6

EXPERIMENTAL EVALUATIONS & RESULTS

6.1 Evaluation Test bed

To evaluate the performance and reliability of the Online Auction App, we conducted tests using real devices and a live Firebase backend environment.

The evaluation was focused on responsiveness, usability, real time data handling, and system behavior under various conditions.

6.1.1 Hardware Used

- Smartphone: Android 11+, 4GB RAM for user testing
- Laptop/PC: Core i5, 8GB RAM for development and monitoring backend operations
- Internet Connection: Stable 10 Mbps Wi-Fi connection

6.1.2 Software Environment

- Flutter SDK: For mobile development.
- Firebase Console: Real time database, authentication, and cloud functions
- Visual Studio Code: Source code editing and debugging.
- Android Studio Emulator: For initial testing and debugging.

6.1.3 Testing Tools

- Manual testing across real Android devices.
- Firebase debug tools and logs for backend behavior monitoring.
- In-app logging to track errors and data flow.

6.1.4 Evaluation Metrics

- Response Time: Time taken to load content and place bids.
- Real time Performance: Accuracy and speed of real-time bid updates.
- User Experience: Ease of navigation, UI responsiveness.
- Error Handling: How the app behaves in error or network failure scenarios.

6.2 Results and Discussion

6.2.1 Functional Performance

- Users were able to sign up, login, and update their profile.
- Sellers could upload items with images and set starting bid prices.
- Buyers were able to place bids that updated instantly across other users' screens using Firebase real-time database.
- The chat system worked without delays, enabling smooth buyer-seller communication.
- Notifications triggered correctly when a new bid was placed or a message was received.

6.2.2 Real time Bidding Evaluation

- Bids updated almost instantly (within 0.5–1 second delay).
- No data loss occurred during simultaneous bid placements.

- Consistent synchronization across devices.

6.2.3 User Feedback

- Easy to use
- Visually appealing and modern.
- Convenient for buying and selling items in an auction format

Minor UI improvements were noted during testing, such as:

- Adding a countdown timer for auction end.
- Improving button placements for small screens.

6.2.4 Performance Under Load

- The app handled 5–10 concurrent users without any slowdown or crash.
- Firebase's real time services scaled automatically without configuration changes.
- No major performance bottlenecks were observed during evaluation.

CHAPTER 7

CONCLUSION AND DISCUSSION

7.1 Strength of this Project

- **User-Friendly Interface:** The app has a clean, intuitive design using Flutter, making it easy to use for both buyers and sellers.
- **Real time Functionality:** Firebase ensures that bidding, chatting, and notifications work in real time, offering a responsive and engaging user experience.
- **Cross-Platform Support:** Developed with Flutter, the app can run on both Android and iOS devices without the need for separate codebases.
- **Secure Authentication:** Firebase Authentication provides safe and reliable login and registration for all users.
- **Interactive Features:** In-app chat and push notifications increase user interaction and improve communication between buyers and sellers.
- **Scalability:** Firebase allows the app to handle multiple users and can be scaled for larger usage in the future.

7.2 Limitations and Future Work

7.2.1 Limitations

- **No Payment Integration:** Currently, the app does not include a built-in payment system, so transactions must be handled externally.
- **Admin Dashboard Missing:** There is no admin panel to manage users, monitor activity, or handle complaints and reports.
- **Limited Auction Controls:** Sellers cannot currently set specific auction time limits or extend auctions.
- **No Rating or Review System:** Buyers and sellers cannot leave feedback, which could help build trust and reputation.

6.2.2 Future Work

- Secure Payment Gateway Integration.
- Admin Panel for system monitoring and management.
- Auction Timer and bidding countdown for each item.
- Review and Rating System for users.
- Dark Mode and accessibility enhancements.
- Web Version of the platform for wider reach.
- These enhancements will make the app more professional.

7.3 Reasons for Failure – If Any

- **Time Management:** Managing time between different modules UI design, backend, testing was difficult, especially in early stages.
- **Learning Curve:** Gaining familiarity with Firebase and real-time databases took time and effort.
- **Limited Testing Devices:** We were only able to test on a few Android devices.

REFERENCES

- D. Gupta, How to Build an Online Auction Application.Benefits, Features, Costs, 2024.
- G. Patil, Online Auction App Development: A Complete Guide, India (Delhi), 2022.
- J. Tiwary, Auction App Development: Unlocking the Future of Online Bidding, India (Mumbai), 30 October, 2023.
- M. Butt, How to Create an Online Real-Time Bidding/Auction App, Mumbai (bendra), 2022.
- Ajay, How to Develop an Application for Online Auctions? Features, Advantages, and Costs, Chennai, 2024.

APPENDICES

List of Appendices

- A0. Copy of Project Registration Form
- A1a. Project Proposal and Vision Document
- A1b. Copy of Proposal Evaluation Comments by Jury
- A2. Requirement Specifications
- A3. Design Specifications
- A4. Other Technical Details
- Test cases
- UI/UX Details
- Coding Standards
- Project Policy
- A5. Flyer & Poster Design
- A6. Copy of Evaluation Comments
- Copy of Evaluation Comments by Supervisor for Project – I Mid Semester Evaluation
- Copy of Evaluation Comments by Jury for Project – I End Semester Evaluation
- Copy of Evaluation Comments by Supervisor for Project – II Mid Semester Evaluation
- Copy of Evaluation Comments by Jury for Project – II Mid Semester Evaluation
- Copy of Evaluation Comments by Jury for Project – II End Semester Evaluation
- A7. Meetings' Minutes
- A8. Research Paper
- A10. Any other

A0. COPY OF PROJECT REGISTRATION FORM

A Photostat or scanned copy should be placed when submitting a document to Project Coordinator. (**Note:** Please remove this line when attach copy that is required)

A1A. PROJECT PROPOSAL AND VISION DOCUMENT

1. Introduction:

The Online Auction System is built with the primary goal of easing the process of conducting auctions for users, which raises the need for constructive non-standard approaches. This application has also enhanced the bidding mechanism, so instead of offering items at fixed prices, bidders can participate in an item's bidding process and buy it for a more competitive price. Another enhancement is the implementation of a user profile management system that allows users to register and manage their profiles securely, an auction module building which encompasses the description of items and the conditions of the bids, and a time-dependent bidding module where users can track the history of the bids and update their status. The ability to communicate directly for price negotiations is facilitated via a chatting system, while the admin side allows for a streamlined management of the auctions, user accounts, and user disputes. Users are also instantly notified about the changes in the auction status and bids, in addition to the fact that the application is optimized for any mobile device which means that it is easy for users to use the application when they are busy doing other tasks.

2. Problem Statement:

Current e-commerce platforms limit interactions with fixed pricing, reducing opportunities for buyers and sellers. The Online Auction App transforms this by offering a dynamic, real-time bidding system, enhancing engagement and maximizing value for both parties.

3. Project Milestones:

- Initial Setup and Planning: Set up the development environment and outline the project's goals and requirements.
- User Registration and Login: Build the basic functionality allowing users to create accounts, log in, and manage their profiles.
- Auction Listing Feature: Enable users to post items for auction with details like start/end times and minimum bids.
- Bidding System: Implement real-time bidding, where users can place and track bids on auction items.
- Chat System: Develop a feature allowing buyers and sellers to communicate directly through the app.
- Admin Panel Development: Create an admin interface for managing users, auctions, and handling disputes.
- Notification System: Add real-time alerts to inform users about bid updates and auction status changes.
- Testing and Quality Assurance: Test the app thoroughly to ensure it works well and fixes any issues found.

- **Launch and Deployment:** Release the app to users, ensuring it's accessible and functioning as expected.
- **Feedback and Improvements:** Collect user feedback and make necessary updates to enhance the app's functionality and user experience.

4. Literature Review:

The Online Auction App seeks to innovate the traditional auction process by leveraging modern technology to enhance user experience, security, and engagement. Below is a review of related literature, existing products, and services in the field.

- **Literature Review :**

Evolution of Online Auctions: Research has shown that online auction platforms, such as eBay, have significantly transformed traditional auctions by allowing global participation. Studies emphasize the importance of user experience and real-time bidding as key factors in user satisfaction.

User Engagement: Literature highlights the role of gamification and real-time notifications in keeping users engaged. Features like bidding updates and personalized messages can improve user retention and activity.

Security Concerns: Numerous studies have focused on the critical need for secure transactions and data protection in online platforms. Strong authentication, encryption, and fraud detection mechanisms are crucial for maintaining user trust.

- **Product Review**

eBay: One of the pioneers in online auctions, eBay offers a vast marketplace with auction and fixed-price options. Its strengths include a wide user base and advanced bidding features, but it can be complex for new users.

Amazon Auctions: This platform provides auction services but lacks the comprehensive auction-specific features found in dedicated auction platforms. Its integration with Amazon's vast marketplace is a significant advantage.

BidSpotter: Specializing in industrial auctions, BidSpotter offers live auction streaming and bidding. Its niche focus is its strength, though it may not cater to general consumer needs.

- **Service Review**

Many platforms now offer mobile apps to facilitate on-the-go bidding. These apps are often praised for their convenience but can suffer from performance issues if not well-optimized.

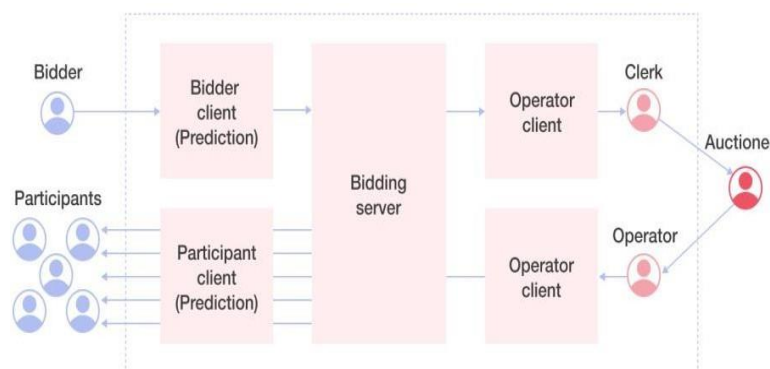
Real-time Notifications and Chat: Services that provide real-time updates and communication between buyers and sellers have been well-received. They enhance transparency and user satisfaction by facilitating direct negotiations and timely updates.

Gap Analysis:

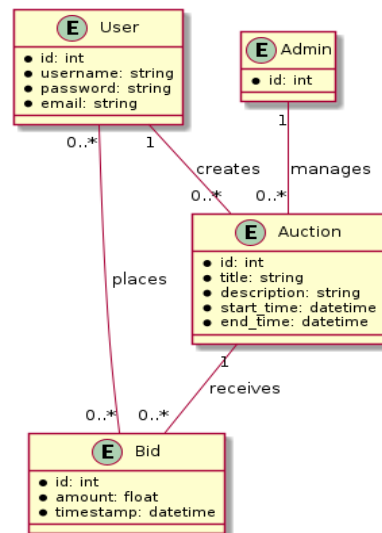
Category	Online Auction System	Competitors (e.g., eBay, Amazon Auctions)	Competitive Advantage
Feature Set			
Real-time Bidding	Offers live updates and bid placement, enhancing user engagement.	Standard feature in most online auction platforms.	Enhanced user engagement through live updates.
User Profile Management	Secure registration and profile management with email verification.	Typically includes secure login and profile customization.	Focused on security and ease of use.
Integrated Chat System	Direct communication between buyers and sellers.	Often limited to post-sale communication; some may lack integrated chat.	Interactive communication boosts negotiation efficiency.
Mobile Compatibility	Optimized for mobile devices, ensuring accessibility on-the-go.	Available, but quality of experience can vary, especially in real-time updates and responsiveness.	Superior mobile experience with optimized performance.
Admin Tools	Robust tools for managing auctions, user accounts, and resolving disputes.	Basic tools for managing listings; advanced features may be restricted to enterprise users.	Comprehensive admin capabilities for efficient management.
Notifications	Real-time alerts for bid updates and auction status changes.	Provided, but the immediacy and relevance of notifications can vary.	More immediate and relevant notifications enhance user experience.
Technology Stack			
Framework	Flutter & Dart: Allows for a highly responsive and aesthetically pleasing cross-platform mobile experience.	Varied tech stacks: Competitors may use different combinations of web and mobile technologies.	Fast, scalable, and secure mobile experience.

Category	Online Auction System	Competitors (e.g., eBay, Amazon Auctions)	Competitive Advantage
Backend Services	Firebase: Ensures real-time database updates, secure authentication, and effective notification handling.	Often rely on proprietary or third-party backend solutions for data handling and real-time interactions.	Reliable and real-time data handling with Firebase.
User Experience			
User-Centric Design	Focus on easy navigation, responsive design, and real-time interactions.	Quality varies; some platforms may not prioritize user experience, leading to complex or outdated interfaces.	Superior user-friendly interface with real-time interactions.
Personalized Notifications	Alerts users to important events in a timely manner.	Can be less immediate, potentially missing critical moments for users.	More engaging through timely notifications.
Real-Time Chat	Enhances negotiation and communication between users.	Often lacks integrated, real-time communication features.	Direct, real-time chat improves user negotiation experience.
Market Positioning			
Target Market	Individual buyers and sellers, small businesses, and niche collectors.	Larger platforms may have a broader reach but can be less personalized.	Niche focus on dynamic, flexible auction experiences tailored to mobile users.
Unique Selling Proposition (USP)	Combines traditional auction features with modern tech, offering a more interactive, mobile-friendly experience.	Established players like eBay have strong brand recognition and extensive user bases.	Carving out a niche in markets seeking more engaging and personalized auction experiences.

Big Picture:



ER Diagram:



Algorithm and Model Findings:

Algorithm:

a) Auction Lifecycle Management

- This algorithm manages the creation, running, and conclusion of auctions.

Steps:

- Initialize auction with start and end times.
- Validate item details(name, category, starting price)
- Continuously monitor for incoming bids during the auction.
- Automatically close auctions after the end time and determine the winner.

b) Bid Validation Algorithm

- This ensures that bids are valid

Steps:

- Check if the auction is active.
- Compare the new bid to the current higher bid.
- If valid, update the current bid and notify participants.

c) Winner Selection Algorithm

- At the end of an auction, this determines the highest bidder as the winner.

Steps:

- Sort of all bids for the auction by bid amount in descending order.
- Select the highest bid.
- Notify the winner and seller.

Model:

The system can be module using the MVC pattern for clarity and scalability.

- Model:**

Handles the business logic and data storage, including auctions, users and bids.

- View:**

User interfaces for bidders and sellers to interact with the system.

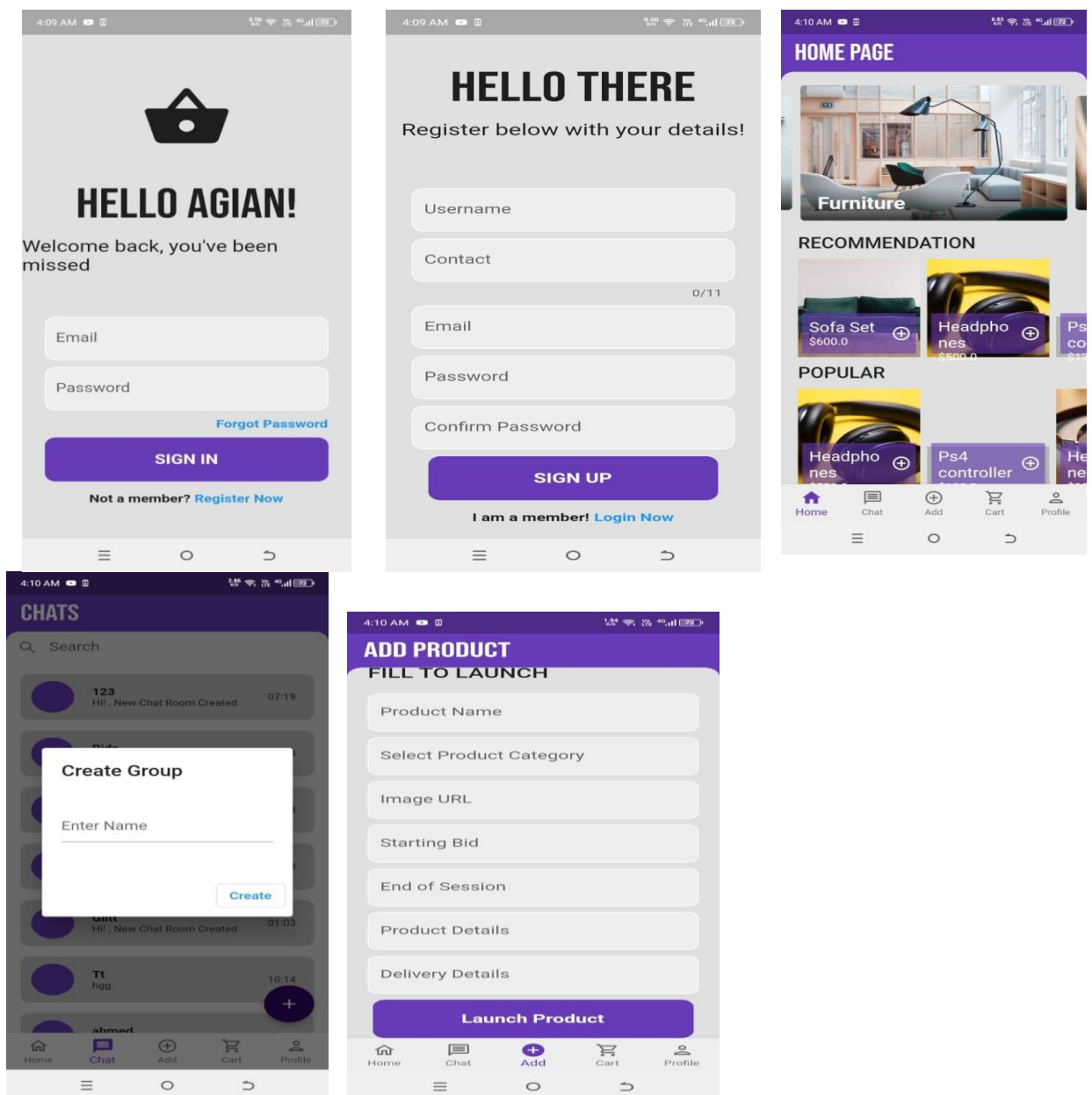
- Controller:**

Coordinates user actions.

Findings:

- **User Engagement:**
Real-time updates on bids create a competitive environment, increasing user activity. Notifications for outbids and auction results enhance user experience.
- **Efficiency:**
Bid validation ensures data integrity, preventing errors like lower bids being accepted. Automated auction management reduces manual intervention.
- **Scalability:**
The modular design allows adding features like payment processing or user reviews without major changes to the system.
- **Potential Challenges:**
Handling high traffic during popular auctions may require server optimization. Ensuring fair play and detecting fraudulent activities.

UI Designs:



References:

- D. Gupta, How to Build an Online Auction Application.Benefits, Features, Costs, 2024.
- G. Patil, Online Auction App Development: A Complete Guide, India (Delhi), 2022.
- J. Tiwary, Auction App Development: Unlocking the Future of Online Bidding, India (Mumbai), 30 October, 2023.
- M. Butt, How to Create an Online Real-Time Bidding/Auction App, Mumbai (bendra), 2022.
- Ajay, How to Develop an Application for Online Auctions? Features, Advantages, and Costs, Chennai, 2024.

A1B. COPY OF PROPOSAL EVALUATION COMMENTS BY JURY

A Photostat or scanned copy should be placed when submitting a document to Project Coordinator. **(Note:** Please remove this line when attach copy that is required)

A2. REQUIREMENT SPECIFICATIONS

Table of Contents

1.	Introduction	4
1.1	Purpose of Document	4
1.2	Intended Audience	4
1.3	Abbreviations	Error! Bookmark not defined.
2.	Overall System Description	6
2.1	Project Background	6
2.2	Problem Statement	6
2.3	Project Scope	7
2.4	Not In Scope	7
2.5	Project Objectives	7
2.6	Stakeholders & Affected Groups	7
2.7	Operating Environment	8
2.8	System Constraints	8
2.9	Assumptions & Dependencies	8
3.	External Interface Requirements	9
3.1	Hardware Interfaces	Error! Bookmark not defined.
3.2	Software Interfaces	9
3.3	Communications Interfaces	9
4.	System Functions / Functional Requirements	10
4.1	System Functions	10
4.2	Use Cases	13
4.2.1	List of Actors	Error! Bookmark not defined.
4.2.2	List of Use Cases	13
4.2.3	Use Case Diagram	14
4.2.4	Description of Use Cases	15
5.	Non - Functional Requirements	Error! Bookmark not defined.
5.1	Performance Requirements	Error! Bookmark not defined.
5.2	Safety Requirements	Error! Bookmark not defined.
5.3	Security Requirements	Error! Bookmark not defined.
5.4	Reliability Requirements	Error! Bookmark not defined.
5.5	Usability Requirements	Error! Bookmark not defined.
5.6	Supportability Requirements	Error! Bookmark not defined.

5.7 User Documentation

6. References

Error! Bookmark not defined.

Error! Bookmark not defined.

1. Introduction

The Online Auction System is built with the primary goal of easing the process of conducting auctions for users, which raises the need for constructive non-standard approaches. This application has also enhanced the bidding mechanism, so instead of offering items at fixed prices, bidders can participate in an item's bidding process and buy it for a more competitive price. Another enhancement is the implementation of a user profile management system that allows users to register and manage their profiles securely, an auction module building which encompasses the description of items and the conditions of the bids, and a time-dependent bidding module where users can track the history of the bids and update their status. The ability to communicate directly for price negotiations is facilitated via a chatting system, while the admin side allows for a streamlined management of the auctions, user accounts, and user disputes. Users are also instantly notified about the changes in the auction status and bids, in addition to the fact that the application is optimized for any mobile device which means that it is easy for users to use the application when they are busy doing other tasks.

1.1 Purpose of Document

The purpose of this document is to provide a detailed Software Requirements Specification (SRS) for the Online Auction App. This document outlines the system's functionality, performance, and operational requirements to ensure the successful implementation and deployment of the application.

1.2 Intended Audience

The Online Auction App is designed for a range of end-users, including: Individual sellers wanting to auction off their items – this could be anything from a car to a collectible Individuals looking to bid on items they are interested in Businesses seeking to expand their revenue stream by offering products for auction Businesses using the app to source products that they may want to integrate into their existing inventory. The app provides value for buyers who are looking for auction items, competitive deals, rare or hard-to-find products and the thrill of bidding. Just as sellers, whether they are small business owners, collectors, or individuals, can benefit from the app by accessing a broader market, garnering competitive bids that can increase their profit margin, and simplifying their selling process.

1.3 Abbreviations

SRS: Software Requirements Specification

FYP: Final Year Project

AI: Artificial Intelligence

SDK: Software Development Kit

IDE: Integrated Development Environment

UI: User Interface

UAT: User Acceptance Testing

CRUD Create, Read, Update, Delete

LAN: Local Area Network

IP: Internet Protocol

MAC: Media Access Control

2. Overall System Description

2.1 Project Background

The online shopping platform is designed to overcome the limitations of traditional e-commerce platforms, which basically offer fixed pricing with no flexibility for customers to negotiate or bid on products role. This application provides a dynamic and interactive environment in which users can engage in real-time trading, giving sellers the opportunity to maximize profits and provide buyers with competitive deals.

To accomplish this, the app uses modern technologies such as Flutter and Dart to create a functional and efficient interface. Google Firebase acts as a database, ensuring data stability, real-time updates, and secure storage. The app has advanced features such as secure email authentication, real-time notifications of new announcements, and chat features, which empower buyers and sellers communicate directly, creating transparency and trust

The app is designed to meet both practical and non-functional needs. Functional requirements include things like store names, checkout options, and user profile management, while non-functional requirements focus on better performance, reliability, and advanced security measures, including password protection and including low latency.

During testing, the app takes several steps to ensure quality and compliance. Unit testing validates individual components, integration testing monitors the interactions between these components, and system testing examines all application functionality. Finally, user acceptance testing (UAT) ensures that the app is global personal expectations, making it user-friendly and effective.

The significance of this project lies in its ability to provide a platform for buyers and sellers to engage in business dynamically. Features such as product categorization, wishlist creation, and personalized messages make it a comprehensive solution for modern auction needs. The Online Auction App bridges the gap between traditional e-commerce and live auction systems, making it an innovative tool for enhancing online trade.

The importance of this project lies in its ability to offer a platform for customers and dealers to interact in enterprise dynamically. Features inclusive of product categorization, wishlist introduction, and customized messages make it a complete solution for current auction wishes. The Online Auction App bridges the distance among conventional e-trade and live auction systems, making it an progressive device for enhancing on line alternate.

2.2 Problem Statement

The problem statement highlights the limitations of present e-trade apps, that specialize in their reliance on fixed pricing fashions. These apps do now not allow buyers to negotiate or bid on merchandise, creating a gap that the Online Auction System pursuits to cope with via introducing real-time bidding.

2.3 Project Scope

The scope of the application describes the main features and functions of the Online Auction App:

- User management: Managing registration, access, and user information.
- Auction List: To include items for auction with information such as start/end times and minimum bids.
- Bidding : Enable real-time bidding and update users on the current bid status.
- Chat system: facilitates communication between buyers and sellers during negotiations.
- Chat bot: Assist users by providing automated responses and enhancing the overall user experience.
- Administration: Managing user accounts, selling and resolving disputes.
- Reports: Sends real-time updates on auctions and bids.
- Mobile Compatibility: Make sure the application is responsive and accessible on all devices.

2.4 Not in Scope

While not explicitly stated, potential Not in Scope items might include:

- Integration with external e-commerce platforms.
- Handling physical delivery of auctioned items.
- Advanced AI-driven price prediction or bidding suggestions.
- Support for desktop platforms.

2.5 Project Objectives

The objectives of the Online Auction System include:

- Providing a secure, user-friendly platform for online auctions.
- Empowering users to buy or sell goods dynamically through bidding.
- Enhancing business opportunities for sellers and offering competitive deals for buyers.
- Facilitating real-time interactions between users for negotiations.
- Ensuring seamless, responsive operation on mobile devices.

2.6 Stakeholders& Affected Groups

Key stakeholders and affected groups include:

- Buyers: Individuals looking for competitive prices and unique items.
- Sellers: Individuals or businesses aiming to reach a broader audience and maximize profits.
- Administrators: Responsible for managing auctions, user disputes, and overall system integrity.
- Developers: Maintaining and updating the app's functionalities.

- Educational Institutions: Using the project as a model for future academic work.

2.7 Operating Environment

The app operates on:

- Mobile Devices: Designed primarily for Android, with potential expansion to iOS.
- Firebase Database: Ensures real-time data updates, secure storage, and robust performance.
- Dart & Flutter Frameworks: Providing a cross-platform development environment optimized for mobile applications.

2.8 System Constraints

- Performance Limitations: Excessive traffic may lead to lag or increased firebase costs.
- Database Dependency : Full reliance on firebase for real-time functionality and storage.
- Platform Restriction: Initially limited to mobile platforms, without desktop support.
- Financial Constraints: Limited resources for scalability and premium hosting services.

2.9 Assumptions & Dependencies

Assumptions:

- Users will have access to stable internet connections for real-time bidding.
- Firebase services will remain affordable and reliable for project needs.
- Buyers and sellers will adhere to platform rules to minimize disputes.

Dependencies:

- Reliance on Firebase for backend operations.
- Compatibility with Android devices during initial implementation.

3. External Interface Requirements

3.1 Software Interfaces

The application interfaces with the following software:

- **Firestore Backend Services:**
 - **Firestore Real time Database:** For storing auction listings, user profiles, and bid updates.
 - **Firestore Authentication:** For user sign-in, sign-up, and email verification.
 - **Firestore Messaging:** For sending notifications to users regarding auctions and bids.
- **External Owners:** Firestore, owned by Google, manages the backend operations, including database hosting, authentication services, and notification delivery.
- **Interface Details:**
 - **Firestore APIs** are used to integrate these services, ensuring seamless communication between the app and the backend.

3.2 Communications Interfaces

The app relies on the following communication interfaces:

- **Network Communication:**
 - **Wi-Fi/Cellular Network:** The app requires an active internet connection to interact with the Firestore services for real-time data sync.
 - **REST APIs:** Used to interact with Firestore for CRUD (Create, Read, Update, Delete) operations.
- **Communication with Devices:**
 - **Push Notifications:** Delivered through Firestore Cloud Messaging, ensuring users receive updates on auctions and bids.
- **Local Area Network (LAN):**
 - Although not explicitly stated, the app can be used over a secure local network if configured, but its primary reliance is on cloud-based communication.

4. System Functions / Functional Requirements

4.1 System Functions

1. User Functionalities

1.1 User Registration:

- Users can create an account by registering with their personal details, including name, email, password, and contact information.
- Email verification is required to activate the account.

1.2 User Login:

- Registered users can log in using their email and password.
- A password reset feature is available for users who forget their credentials.

1.3 Homepage Access:

- After successful login, users are directed to a personalized homepage displaying auction highlights, popular items, and ongoing bids.

1.4 Add Bids:

- Users can view auction items and place bids.
- The system displays the current highest bid and allows users to input their own bid.
- Users receive notifications when they are outbid.

1.5 View Items by Category:

- Users can browse items organized into categories and subcategories for easier navigation.

1.6 Post an Auction:

- Users can post their own auction by providing item details, images, starting bid, and auction duration.
- The auction will be visible to all users once approved.

1.7 Search Product Category:

- Users can search for products by category or keywords using a dynamic search bar.

1.8 Chat System:

- A chat system enables users to communicate with one another in real-time.

- Public chat rooms and private messaging are supported.

1.9 Send Personal Messages:

- Users can send direct messages to other users to discuss items or negotiate terms.
- Messages are accessible through the user's inbox.

2. Admin Functionalities

2.1 Manage Users:

- Admin can view and manage user accounts.
- Admin can edit user details and track user activity.

2.2 Manage Items:

- Admin can view, edit, or delete auction items as necessary.
- Items can be approved or rejected based on compliance with platform policies.

2.3 Manage Categories:

- Admin can create, edit, and delete product categories and subcategories.
- Categories are structured to improve item discoverability.

2.4 Block User:

- Admin can block users who violate platform rules.
- Blocked users cannot log in or participate in auctions.

2.5 Administrative Functions:

- Admin have access to system analytics and reports, including user engagement, auction performance, and revenue statistics.
- Admin can configure system-wide settings, such as notifications and auction rules.

3. Additional Features

3.1 Email Notifications:

- Users and admin receive email notifications for important updates, such as bid status, auction deadlines, and account activities.

3.2 Pop-Up Notifications:

- Real-time pop-up notifications alert users to outbids, auction status changes, or new messages.

3.3 Delivery Information:

- Sellers can provide delivery details for auctioned items.
- Buyers can track the delivery status through the app.

	Functions	Category	Attribute	Details & Boundary Constraints
R1.1	Record the underway sale – the items purchased	Evident	System Response time	Price listing within 3 seconds Availability agreement in less than 10 sec
R1.2	Reduce inventory quantities when a sale is committed	Hidden	Concurrent user load	Supports up to 100 concurrent users without performance degradation.
...	Manage user profiles securely	Hidden	Security	Enforce password complexity and encryption. - Multi-factor authentication.
R1.3	Facilitate competitive bidding for items b	Evident	User Interaction	Real-time updates on the bidding status with sub-second latency.
R1.4	Notify users about changes in auction status or bids	Evident	Real-Time Notification	Notification delivery within 2 seconds of change.
R1.5	Track bid history and update bid status	Evident	Data Accuracy	Bid history is accurate and updated within 1 second of bid placement.
R1.6	Enable direct communication for price negotiations	Evident	Communication response	Chat messages delivered within 1 second.
R1.8	Streamline admin management of auctions and user disputes	Hidden	Admin Efficiency	Admin actions reflected within 5 seconds of operation.
R1.9	Optimize application for mobile devices	Frill	Compatibility	Application responsive and operational on devices with screens as small as 4 inches.

5. System Attributes/ Nonfunctional Requirements

4.1 Performance Requirements

- The application will perform with minimal lag.

- Real time database ensures instant updates and synchronization.

4.2 Reliability

- Ensure data consistency across all users and sessions.

4.3 Security

- Password protection and encryption are implemented to safeguard user data.

Attribute	Details and Boundary Constraints	Category
Response Time	The system should complete important tasks, like placing bids or viewing updates, within 5 seconds.	Optional
Concurrent User Load	The app must handle at least 10 users at the same time without slowing down.	Mandatory
Data Availability	The system should be up and running 99.9% of the time to ensure users can access it whenever needed.	Mandatory
User Interface	The app should be easy to use and work well on different devices.	Optional
Security Measures	Use passwords, encryption, and secure communication to keep user data safe.	Mandatory
Expansion Flexibility	The system should allow easy addition of new features and handle more users as needed.	Optional

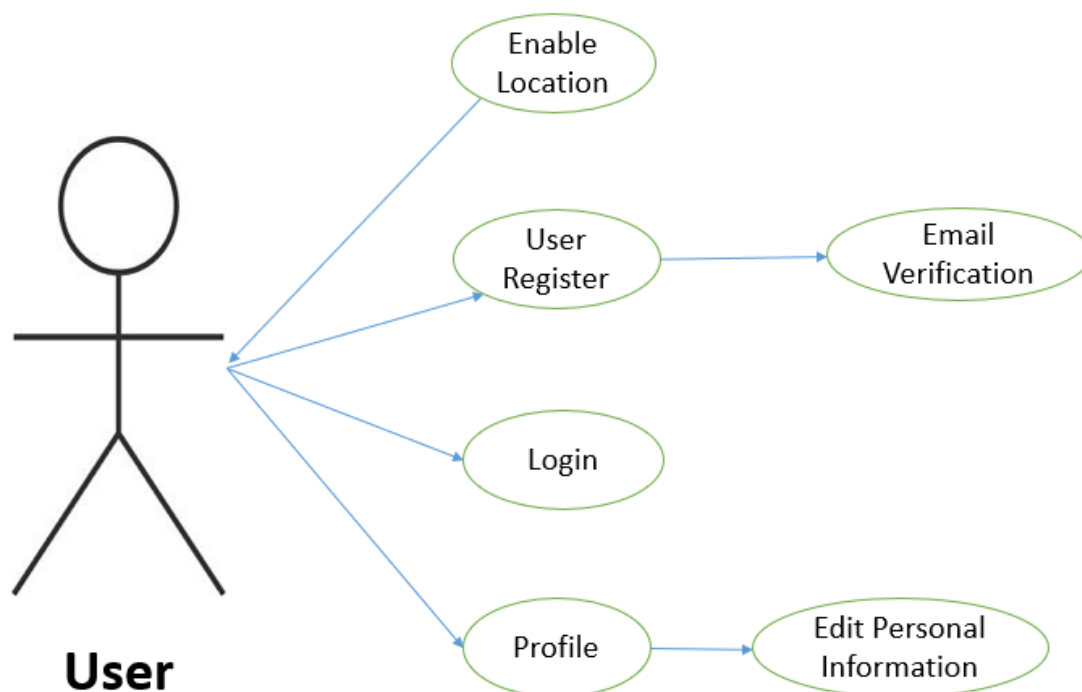
4.2 Use Cases

4.2.1 List of Use Cases

- Enable Location:
The user grants location permissions to the system for the location-based features.
- User Registration:
The user creates an account by entering their details and initiating the registration process.

- Email Verification:
After registration, the system sends a verification email, which the user must validate to activate their account.
- Login:
The user provides credentials (email and password) to authenticate and access their account.
- Profile:
The user views their account details, including personal information and other settings.
- Edit Personal Information:
The user updates their account details, such as name, contact information, or email.

4.2.2 Use Case Diagram



4.2.3 Description of Use Cases

1. Enable User

Name	Enable User
Actors	User
Purpose	To allow the user to enable location services to enhance the functionality of application.
Description	The user can enable location services to receive location specific features, such as displaying nearby auction items or arranging local deliveries.
Cross References	None
Pre-Conditions	The user must have a registered account and be logged in.
Successful Post-Conditions	The location services are enabled, and the application can use the user's location to provide relevant features.
Failure Post-Conditions	Location services remain disabled, and the user cannot access location specific features.

2. User Register

Name	User register
Actors	User
Purpose	To allow a new user to create an account in the system.
Description	The user enters personal information, such as name, email, and password, to register on the platform.
Cross References	Email verification
Pre-Conditions	The user must not have an existing account.
Successful Post-Conditions	A new account is created, and the user receives an email for verification.
Failure Post-Conditions	The account is not created, and the user must try registered again.

3. Email Verification

Name	Email verification
Actors	User
Purpose	To confirm the user's email address to activate the account.
Description	After registration, the user receives a verification email. Clicking the link in the email confirms the user's email address.
Cross References	User Register
Pre-Conditions	The user must have completed the registration process.

Successful Post-Conditions

The user's email is verified, and the account is activated.

Failure Post-Conditions

The email remains unverified, and the user cannot fully access the platform.

4. Login

Name

Login

Actors

User

Purpose

To allow the user to access their account.

Description

The user provides their email and password to log into their account.

Cross References

None

Pre-Conditions

The user must have a registered and verified account.

Successful Post-Conditions

The user is logged in and can access their account.

Failure Post-Conditions

The user remains logged out, and they must retry logging in.

5. Profile

Name

Profile

Actors

User

Purpose

To allow the user to view and manage their personal information.

Description

The user can access their profile to see personal details, such as name, email, and account settings.

Cross References

Edit Personal Information

Pre-Conditions

The user must be logged in.

Successful Post-Conditions

The user can view their profile details.

Failure Post-Conditions

The user cannot view their profile, and they must retry accessing it.

6. Edit Personal Information

Name

Edit Personal Information

Actors

User

Purpose

To allow the user to update their personal details.

Description

The user can edit personal information, such as their name, email, or password, through their profile.

Cross References

Profile

Pre-Conditions

The user can edit personal information, such as their name, email, or password, through their profile.

Successful Post-Conditions

The user's personal information is updated

Failure Post-Conditions

successfully.

The changes are not saved, and the user must retry editing their information.

4.2.5 Typical Course Events

Actor Action	System Response
User registers for a new account	The system validates the input, sends an email verification, and upon successful validation, creates a new user account.
User logs in with credentials	The system verifies the credentials and grants access to the user's dashboard if the credentials are correct.
User views available auctions	The system retrieves and displays a list of active auctions categorized by type.
User places a bid on an item	The system checks if the bid is higher than the current highest bid, updates the bid status, and notifies the user of the bid status.
User posts a new auction	The system validates the auction details, posts the auction in the relevant category, and notifies the user of a successful post.
Admin manages user accounts	The system allows the admin to view, edit, block, or delete user accounts based on administrative actions.
User sends a message in the chat system	The system delivers the message to the intended recipient and notifies both parties of new messages.
User receives a notification on bid update	The system sends a real-time notification to the user regarding updates on bids they are interested in.
Admin manages auction listings	The system enables the admin to edit, approve, or remove auction listings and resolves any disputes arising from auctions.
User logs out	The system logs the user out, ensuring session termination and data security.

References

- D. Gupta, How to Build an Online Auction Application Benefits, Features, Costs, 2024.

- G. Patil, Online Auction App Development: A Complete Guide, India (Delhi), 2022.
- J. Tiwary, Auction App Development: Unlocking the Future of Online Bidding, India (Mumbai), 30 October, 2023.
- M. Butt, How to Create an Online Real-Time Bidding/Auction App, Mumbai (bendra), 2022.
- Ajay, How to Develop an Application for Online Auctions Features, Advantages, and Costs, Chennai, 2024.

A3. DESIGN SPECIFICATIONS

Term	Description
Online Auction	A digital platform where users can bid on items, with the highest bidder winning the item at the close.
Auction Engine	The core system managing auction functionalities such as item listings, bidding processes, and timers.
User Roles	
Bidder	Individuals participating in auctions by placing bids.
Seller/Auctioneer	Users listing items for auction.
Admin	System managers handling user management, disputes, and auction oversight.
Timed Auction	An auction type where bids are accepted for a fixed duration.
Reserve Auction	An auction type where the item is sold only if bidding meets a preset reserve price.
"Buy Now" Option	A feature allowing buyers to purchase items at a fixed price without bidding.
Real-Time Bidding	A system enabling instantaneous bid placement and updates, ensuring seamless interactions.
Chat Module	An integrated feature allowing direct communication between buyers and sellers for negotiations.
User Authentication	A secure login mechanism, including encrypted passwords and potentially two-factor authentication.
Notification System	Alerts users about bid status, auction closures, and system updates in real-time.
Data Encryption	A method of safeguarding user data and transactions against unauthorized access.
Caching Mechanisms	Technology like Redis used to speed up data retrieval for active auctions and frequent queries.
Fraud Detection	Systems and algorithms to identify and mitigate fraudulent activities, such as fake listings or shill bidding.
COD (Cash on Delivery)	A payment option where buyers pay upon receiving the product, often integrated with logistics systems.
Payment Gateway	Third-party services facilitating secure online transactions, including COD and escrow payments.
Accessibility Standards	Compliance with guidelines like WCAG to ensure the app is usable by individuals with disabilities.
Regulatory Compliance	Adherence to laws and regulations governing online auctions, including data protection and e-commerce.
Scalability	The app's ability to handle increased traffic and user growth without performance issues.
Dispute Resolution	Mechanisms to address and resolve conflicts between buyers and

Term	Description
	sellers effectively.

Table of Contents

1.	Introduction	4
1.1	Purpose of Document	4
1.2	Intended Audience	4
1.3	Abbreviations	Error! Bookmark not defined.
2.	Overall System Description	6
2.1	Project Background	6
2.2	Problem Statement	6
2.3	Project Scope	7
2.4	Not In Scope	7
2.5	Project Objectives	7
2.6	Stakeholders & Affected Groups	7
2.7	Operating Environment	8
2.8	System Constraints	8
2.9	Assumptions & Dependencies	8
3.	External Interface Requirements	9
3.1	Hardware Interfaces	Error! Bookmark not defined.
3.2	Software Interfaces	9
3.3	Communications Interfaces	9
4.	System Functions / Functional Requirements	10
4.1	System Functions	10
4.2	Use Cases	13
4.2.1	List of Actors	Error! Bookmark not defined.
4.2.2	List of Use Cases	13
4.2.3	Use Case Diagram	14
4.2.4	Description of Use Cases	15
5.	Non - Functional Requirements	Error! Bookmark not defined.
5.1	Performance Requirements	Error! Bookmark not defined.
5.2	Safety Requirements	Error! Bookmark not defined.
5.3	Security Requirements	Error! Bookmark not defined.
5.4	Reliability Requirements	Error! Bookmark not defined.
5.5	Usability Requirements	Error! Bookmark not defined.
5.6	Supportability Requirements	Error! Bookmark not defined.
5.7	User Documentation	Error! Bookmark not defined.
6.	References	Error! Bookmark not defined.

Introduction

The Online Auction System is built with the primary goal of easing the process of conducting auctions for users, which raises the need for constructive non-standard approaches. This application has also enhanced the bidding mechanism, so instead of offering items at fixed prices, bidders can participate in an item's bidding process and buy it for a more competitive price. Another enhancement is the implementation of a user profile management system that allows users to register and manage their profiles securely, an auction module building which encompasses the description of items and the conditions of the bids, and a time-dependent bidding module where users can track the history of the bids and update their status. The ability to communicate directly for price negotiations is facilitated via a chatting system, while the admin side allows for a streamlined management of the auctions, user accounts, and user disputes. Users are also instantly notified about the changes in the auction status and bids, in addition to the fact that the application is optimized for any mobile device which means that it is easy for users to use the application when they are busy doing other tasks.

Purpose of Document

The purpose of this document is to provide comprehensive Software Design Specifications (SDS) for the Online Auction App. Outlining the design methodology, architecture and detailed design considerations. This document is intended to guide the development team and serve as a reference for stakeholders.

Intended Audience

This document is intended for

- Development Team members
- Project managers
- System architectures
- QA testers
- Stakeholders and supervisors

Document Convention

The font: Arial

Font size: 12

Project Overview

The Online Auction App is a digital platform designed to make buying and selling items easier and more dynamic. Unlike regular e-commerce apps with fixed prices, this app lets users bid on products, giving buyers a chance to get better deals and sellers an opportunity to earn more through competitive bidding.

It includes features like real-time updates on bids, user profiles to manage activities, and a chat system for direct communication between buyers and sellers. The app is secure, ensuring user data and transactions are safe, and works smoothly on mobile devices so users can participate in auctions anytime, anywhere. Administrators help

manage the system, handle disputes, and ensure everything runs smoothly.

Overall, this app bridges the gap between traditional e-commerce and live auction systems, providing a modern and efficient solution for online trading. Whether you're a buyer looking for bargains or a seller wanting to reach a wider audience, this app offers everything you need for a seamless auction experience

Scope

- The Online Auction App is designed to create a modern, interactive platform for buying and selling items through real-time bidding. It allows users to register, create profiles, and post items for auction while providing a secure and user-friendly environment. Buyers can browse auctions, place bids, and receive instant updates about their bid status. Sellers benefit from reaching a wider audience and maximizing profits through competitive bidding.
- The app includes built-in chat features, enabling buyers and sellers to communicate directly, making negotiations and transactions smoother. It is optimized for mobile devices, ensuring that users can participate in auctions anytime and anywhere. Additionally, the app incorporates strong security measures to protect user data and transactions, such as encrypted communication and secure login processes. Powered by Firebase, the app ensures real-time updates and scalability, allowing it to handle a growing number of users and auctions seamlessly.
- The scope of this project focuses on delivering a reliable, accessible, and engaging online auction experience, bridging the gaps in traditional e-commerce platforms and empowering both buyers and sellers with innovative tools and features.

Design Considerations

Here's a tailored version of the text for your project on an **online auction app**:

4.1.1 Problem Identification and Preliminary Considerations

This section outlines the critical issues and requirements that need to be addressed before creating a comprehensive design solution for the online auction app. These considerations serve as the foundation for a robust, user-centric system design.

User Accessibility and Experience

- Ensure the app is intuitive and user-friendly for both buyers and sellers.
- Provide seamless navigation, from account creation to bidding and selling processes.
- Optimize the app for different devices (mobile, tablet, and desktop) and platforms (iOS, Android, and web browsers).

Security and Trust

- Implement secure authentication (e.g., two-factor authentication).
- Safeguard user data and financial transactions through encryption and compliance with data protection laws.
- Include features like verified user badges, feedback systems, and dispute resolution mechanisms to build trust.

Bidding and Auction Processes

- Support multiple auction types, such as timed auctions, reserve auctions, and "Buy Now" options.
- Real-time bidding updates with notifications for outbids and auction endings.
- Prevent fraudulent practices, such as shill bidding or fake listings, with automated monitoring.

Payment and Transaction Management

- Provide an option for sellers to enable or disable COD for their listings based on their preferences.
- Clearly indicate to buyers during checkout whether COD is available for their selected items.
- Specify delivery regions or conditions where COD is applicable.

Scalability and Performance

- Design the system to handle a growing user base and large volumes of concurrent bids without performance degradation.

- Employ efficient database management for real-time data updates.
- Optimize load balancing and caching mechanisms to reduce latency.

Regulatory and Compliance Requirements

- Adhere to regional and international e-commerce and auction laws.
- Ensure taxation rules are applied correctly based on user locations.
- Address legal requirements for selling restricted or regulated items.

Customer Support and Feedback

- Provide multiple channels for customer support (chat, email, FAQs, etc.).
- Enable users to report issues directly through the app.
- Regularly collect user feedback to drive continuous improvement.

Marketing and Growth

- Include features for promotional offers, such as discounted listing fees or special bidding events.
- Integrate social media sharing to increase user engagement and app visibility.
- Use analytics to track user behavior and improve marketing strategies.

Assumptions and Dependencies

4.2.1 Design-Specific Assumptions and Dependencies

Dynamic Real-Time Updates

- **Assumption:** The app must support real-time updates for auction events (e.g., bidding activity, auction expiration).
- **Dependency:** The design relies on WebSocket technology or similar real-time communication protocols to ensure low-latency updates. The system must scale to handle concurrent users during high-traffic auctions.

Scalable Modular Architecture

- **Assumption:** The system will need to support modular features like different auction types (e.g., timed, reserve) and payment methods, including COD and escrow services.
- **Dependency:** The design assumes a microservices architecture, where components such as the bidding engine, payment processing, and notifications are independently scalable.

Customizable User Interfaces

- **Assumption:** The app will offer different user roles (buyers, sellers, and admins), each with tailored dashboards and workflows.

- **Dependency:** The design depends on frameworks (e.g., React, Angular) that allow responsive, role-specific, and customizable UI elements.

Geographical and Regional Constraints

- **Assumption:** The system must accommodate region-specific features, such as language, currency, and tax rules.
- **Dependency:** The design incorporates APIs for localization, exchange rates, and tax calculations. Additionally, the system must dynamically adapt based on the user's location.

Secure Payment and Delivery Integration

- **Assumption:** For payments, COD will require integration with logistics providers to track deliveries and confirm transactions securely.
- **Dependency:** The design assumes APIs for delivery tracking, payment reconciliation, and secure data transfer between the app and third-party services.

Fraud and Dispute Management

- **Assumption:** Users will expect mechanisms to detect fraudulent activities and resolve disputes efficiently.
- **Dependency:** The design assumes the integration of AI-driven fraud detection tools and a scalable dispute resolution module for automated and manual review.

Flexibility in Feature Addition

- **Assumption:** New features (e.g., social sharing, in-app chat, or AI recommendations) may need to be added post-launch without disrupting existing workflows.
- **Dependency:** The system must employ a flexible plugin or API-based structure to allow feature expansion with minimal downtime.

Third-Party Services

- **Assumption:** Critical features like notifications, analytics, and performance monitoring will rely on third-party services (e.g., Firebase, Mixpanel, Twilio).
- **Dependency:** The design assumes stable APIs and continued support from these services, with fallback mechanisms in place for potential service interruptions.

Compliance with Accessibility Standards

- **Assumption:** The app must comply with accessibility guidelines to cater to users with disabilities.

- **Dependency:** The design depends on accessible design principles (e.g., WCAG) and tools for testing accessibility compliance during the development process.

Support for High Availability and Disaster Recovery

- **Assumption:** The app should remain operational during high traffic and recover quickly from unexpected downtimes.
- **Dependency:** The design assumes the use of cloud services with built-in redundancy, auto-scaling, and disaster recovery mechanisms.

Risks and Volatile Areas

Emerging User Requirements

- **Source of Change:** Users may demand additional features like advanced filtering, new auction types, or expanded payment options (e.g., cryptocurrency, BNPL).
- **Impact:** Requires iterative development and system modularity to integrate new features without disrupting core functionalities.
- **Mitigation:**
 - Design the system with a modular architecture to allow independent addition of new features.
 - Use Agile development methodologies to respond to changing user requirements in short cycles.

Technological Advancements

- **Source of Change:** Adoption of new technologies (e.g., AI for fraud detection, AR for virtual item previews, blockchain for transparent bidding).
- **Impact:** Upgrades or overhauls in existing systems may be necessary to stay competitive.
- **Mitigation:**
 - Leverage APIs and cloud-based services that allow easy integration with cutting-edge technologies.
 - Regularly review technological trends to proactively plan for updates.

Regulatory and Compliance Changes

- **Source of Change:** Updates to e-commerce, auction, or data protection laws across regions.
- **Impact:** Non-compliance could lead to legal penalties or service restrictions in specific regions.
- **Mitigation:**
 - Maintain a legal and compliance team to monitor regulatory updates.
 - Design the system to support region-specific configurations for tax laws, restricted items, and data handling.

Scalability Demands

- **Source of Change:** A sudden increase in users or high-profile auctions leading to system overload.
- **Impact:** Performance degradation, slow bidding updates, or system crashes could harm user trust.
- **Mitigation:**
 - Utilize cloud infrastructure with auto-scaling capabilities to handle peak loads.
 - Employ load testing during development to ensure system stability under high traffic.

Fraud and Security Threats

- **Source of Change:** Increasing sophistication in fraud techniques (e.g., fake accounts, shill bidding, item misrepresentation).
- **Impact:** Loss of user trust, financial losses, and reputational damage.
- **Mitigation:**
 - Integrate AI-driven fraud detection and monitoring tools.
 - Regularly update security protocols and conduct penetration testing.

Competition and Market Dynamics

- **Source of Change:** Competitors introducing innovative features or offering lower transaction fees.
- **Impact:** Risk of user attrition and reduced market share.
- **Mitigation:**
 - Monitor competitors and user feedback to stay ahead in innovation.
 - Implement loyalty programs and promotional campaigns to retain users.

Logistics and Payment Ecosystem Changes

- **Source of Change:** Dependency on third-party logistics providers and payment gateways that may revise APIs or terms.
- **Impact:** Disruptions in payment processing or COD fulfillment.
- **Mitigation:**
 - Partner with multiple logistics providers and payment gateways to reduce single points of failure.
 - Periodically update integrations to align with third-party changes.

Contingency Paths for Changes

- Maintain a flexible microservices architecture that allows isolated updates without affecting the entire system.
- Use feature toggles to enable or disable new features in response to unexpected challenges.

- Develop a rollback plan for major updates to revert to stable versions if unforeseen issues arise.

System Architecture

This section provides a high-level overview of the **online auction app**, describing how its functionality and responsibilities are divided among various components and subsystems. The architecture ensures scalability, maintainability, and efficient collaboration between components to deliver a seamless user experience.

High-Level Overview

The system follows a **modular and micro services-based architecture**, ensuring that each component operates independently while interacting with others through defined interfaces. It is divided into the following layers:

1. Presentation Layer

- **Purpose:** User interaction via mobile and web applications.
- **Components:**
 - Mobile App (iOS/Android)
 - Web Interface (React/Angular-based)
- **Responsibilities:**
 - Provide an intuitive UI for buyers, sellers, and admins.
 - Handle real-time updates for bids, notifications, and auction statuses.
 - Ensure responsiveness and accessibility across devices.

2. Business Logic Layer

- **Purpose:** Manage core functionalities and workflows of the auction system.
- **Components:**
 - **Auction Engine:** Manages auction creation, bidding processes, and timer functionalities.
 - **User Management Service:** Handles registration, login, roles, and profile management.
 - **Payment Module:** Manages payment methods, including COD, escrow, and online payments.
 - **Notification Service:** Sends real-time alerts (e.g., outbids, auction end) via email, SMS, and push notifications.
- **Responsibilities:**
 - Ensure real-time bid updates.
 - Validate bids, manage auction rules, and resolve conflicts.
 - Integrate fraud detection algorithms.

3. Data Layer

- **Purpose:** Manage storage and retrieval of system data.
- **Components:**
 - **Relational Database (e.g., PostgreSQL):** For structured data like user profiles, auction details, and transaction records.
 - **NoSQL Database (e.g., MongoDB):** For unstructured data like logs, notifications, and analytics.

- Cache Layer (e.g., Redis): For high-speed access to frequently queried data, like active auctions and bids.
- **Responsibilities:**
 - Maintain data consistency and integrity.
 - Provide APIs for seamless data retrieval and updates.
- 4. **Integration Layer**
 - **Purpose:** Facilitate communication with third-party services.
 - **Components:**
 - **Payment Gateways:** Support secure transactions for COD and online payments.
 - **Logistics APIs:** Handle shipping, tracking, and COD verification.
 - **Analytics and Monitoring Tools:** Collect and analyze user activity and system performance.
 - **Responsibilities:**
 - Manage external dependencies like payment and delivery services.
 - Ensure API reliability and fallback mechanisms for third-party failures.
- 5. **Infrastructure Layer**
 - **Purpose:** Provide a robust environment for system deployment and scalability.
 - **Components:**
 - Cloud Hosting (e.g., AWS, Azure, GCP): For deployment, scalability, and disaster recovery.
 - Load Balancers: Distribute traffic across servers to avoid overload.
 - Monitoring Tools (e.g., Prometheus, Grafana): Track system performance and uptime.
 - **Responsibilities:**
 - Enable auto-scaling to handle peak traffic during high-profile auctions.
 - Provide disaster recovery and backups to ensure data safety.

Interaction Between Components

1. **User Interaction Flow:**
 - Users interact with the app via the Presentation Layer.
 - User requests (e.g., placing a bid, viewing auctions) are processed by the Business Logic Layer.
 - Data retrieval and updates occur in the Data Layer, ensuring minimal latency through caching.
 - Notifications are sent via the Notification Service for key events (e.g., auction updates, bid success).
2. **Auction Process Flow:**
 - Sellers create auctions via the UI, processed by the Auction Engine.
 - Buyers place bids in real-time, validated and updated in the system through the Business Logic Layer.
 - Auction results are finalized, and payment/fulfillment processes are initiated using Payment and Logistics APIs.

3. COD Workflow:

- Buyers select COD at checkout, triggering the Logistics API for order confirmation.
- Delivery updates are tracked, and cash is reconciled via logistics partners before releasing funds to the seller.

5.3 Scalability and Extensibility

- The architecture supports future feature additions (e.g., new auction types, advanced analytics) via modular components.
- High scalability is achieved through containerized microservices (e.g., using Docker and Kubernetes).
- APIs ensure seamless integration with new third-party services as the system evolves.
- System Level Architecture

The system-level architecture for the Online Auction System is designed to provide a modular, maintainable, and scalable structure. It decomposes the system into subsystems and components, detailing their roles, relationships, and interfaces.

System Decomposition into Elements

1. User Interface Layer:

- Provides the interface for users (bidders, auctioneers, and admins) to interact with the system.
- Includes web and mobile applications, ensuring accessibility across devices.

2. Application Logic Layer:

- Responsible for implementing the business logic and core functionalities of the system, such as bid validation, auction management, and notifications.
- Contains modules like:
 - **User Management:** Handles user authentication and profile management.
 - **Auction Module:** Manages the creation, display, and updating of auctions.
 - **Bid Module:** Processes bids and maintains bid histories.
 - **Notification System:** Sends real-time notifications to users.

3. Data Access Layer:

- Interacts with the database to store and retrieve data related to users, auctions, bids, and notifications.

4. Database Layer:

- Stores all persistent data, including user profiles, auction details, bids, chat messages, and system logs.

5. External Systems:

- Payment Gateway: Integrates with external services for secure payment processing (optional, if included in the app).
- Notification Services: Interfaces with external APIs for sending emails or push notifications.

Software Architecture

The software architecture for the Online Auction System is designed with a layered approach for simplicity and modularity.

Layers of Architecture

1. User Interface Layer

- Manages interactions with the users (bidders, auctioneers, admins).
- Example technologies: React.js, Angular, or mobile frameworks like Flutter.

2. Middle Tier (Business Logic)

- Handles the main business logic.
- Manages bid validation, auction rules, and real-time updates.
- Technologies: Node.js, Java Spring Boot, or .NET Core.

3. Data Access Layer

- Interfaces between the application and the database.
- Implements ORM frameworks like Hibernate or Sequelize.

4. Database Layer

- Stores all application data persistently.
- Firebase

Design Strategy

The **design strategy** for the online auction app is centered around scalability, modularity, and user-centric functionality. This section outlines the key abstractions, mechanisms, and decision-making processes that shape the system's high-level organization while addressing considerations such as extensibility, reuse, interface paradigms, data management, and concurrency.

Future System Extension or Enhancement

Strategy:

- Use a **modular architecture** where each component (e.g., bidding engine, payment gateway, notification system) operates independently.
- Adopt **microservices** for core functionalities to enable easy addition or replacement of features.
- Ensure **API-driven development**, allowing integration with new third-party services without disrupting the system.

Reasoning:

- User needs and market trends evolve, necessitating a flexible system that supports new auction types (e.g., reverse auctions) or payment methods (e.g., crypto currencies).
- The micro services approach ensures isolated updates and prevents downtime.

Trade-offs:

- Increased complexity in managing inter-service communication and dependencies.
- Potential latency in distributed systems compared to monolithic architectures.

System Reuse

Strategy:

- Design reusable components for common functionalities like user authentication, payment processing, and notifications.
- Employ shared libraries and frameworks for tasks such as logging, error handling, and security.

Reasoning:

- Reuse reduces development effort for future projects or additional platforms (e.g., mobile and web).
- Standardized components improve system consistency and reduce maintenance overhead.

Trade-offs:

- Additional effort is required upfront to ensure components are generic and well-documented.
- Customization for specific use cases may require extending the reusable components.

User Interface Paradigms

Strategy:

- Use a **responsive design** paradigm to ensure a seamless experience across devices.
- Implement **role-based interfaces** tailored to buyers, sellers, and administrators.
- Focus on **real-time interactivity** with Web Socket or Server-Sent Events (SSE) for updates during auctions.

Reasoning:

- The app serves diverse user roles and device types, making adaptable UIs essential.
- Real-time feedback during bidding increases user engagement and ensures competitive fairness.

Trade-offs:

- Real-time UI updates require robust backend systems and increased network demands.
- Implementing role-specific interfaces can increase design complexity.

Data Management

Strategy:

- Use a combination of **relational databases** (e.g., PostgreSQL) for structured data (e.g., user profiles, auction details) and **No SQL databases** (e.g., MongoDB) for unstructured data (e.g., logs, analytics).
- Employ **caching mechanisms** (e.g., Redis) for frequently accessed data like active auctions and bid histories.
- Design the system to support **data partitioning and replication** for high availability.

Reasoning:

- A hybrid database approach optimizes for performance and scalability.
- Caching reduces latency and ensures real-time data availability during critical auction moments.

Trade-offs:

- Managing hybrid databases introduces additional complexity.
- Consistency issues may arise between cached data and the primary database, requiring robust synchronization mechanisms.

Concurrency and Synchronization

Strategy:

- Use **optimistic locking** or **versioning** for handling concurrent updates in critical operations like placing bids.
- Employ **event-driven architecture** with message queues (e.g., RabbitMQ, Kafka) to manage high-concurrency scenarios.
- Design backend services to be **stateless**, enabling horizontal scaling for increased concurrent user loads.

Reasoning:

- Auctions involve real-time interactions, making concurrency management crucial to avoid race conditions or data corruption.
- Event-driven systems decouple components, improving scalability and reliability under heavy loads.

Trade-offs:

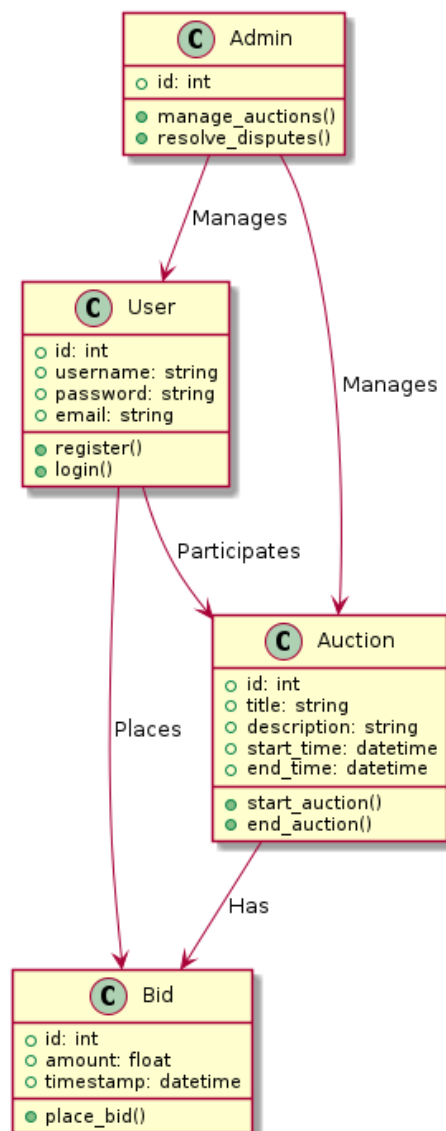
- Increased complexity in managing distributed transactions.
- Real-time bid synchronization requires significant infrastructure investment.

Detailed System Design

Design

Class

Diagram



Database Design

7.2.1 Data 1: User

Name	User
Alias	Bidder, Auctioneer

Name User

Content Description ID + Username + Password + Email + Role + Created_At

Table: User

Column Name	Description	Type	Length	Nullable	Default Value	Key Type
ID	Unique identifier for user	Integer	-	No	Auto-increment	PK
Username	Name of the user	String	100	No	NULL	
Password	Encrypted password	String	255	No	NULL	
Email	User's email address	String	150	No	NULL	
Role	Role of the user (bidder/seller)	String	50	No	'Bidder'	
Created_At	Account creation timestamp	DateTime	-	No	CURRENT_TIMESTAMP	

7.2.2 Data 2: Admin

Name Admin

Alias System Manager

Where-used/how-used Used for managing users, auctions, and system disputes.

Content Description ID + Username + Password + Email + Created_At

Table: Admin

Column Name	Description	Type	Length	Nullable	Default Value	Key Type
ID	Unique identifier for admin	Integer	-	No	Auto-increment	PK
Username	Name of the admin	String	100	No	NULL	
Password	Encrypted password	String	255	No	NULL	

Column Name	Description	Type	Length	Nullable	Default Value	Key Type
Email	Admin's email address	String	150	No	NULL	
Created_At	Account creation timestamp	DateTime	-	No	CURRENT_TIMESTAMP	

7.2.3 Data 3: Auction

Name	Auction
Alias	Item, Listing
Where-used/how-used	Used for creating auctions, placing bids, and notifications.
Content Description	ID + Title + Description + Start_Time + End_Time + Starting_Bid + Current_High_Bid + User_ID + Created_At

Table: Auction

Column Name	Description	Type	Length	Nullable	Default Value	Key Type
ID	Unique identifier for auction	Integer	-	No	Auto-increment	PK
Title	Title of the auction item	String	200	No	NULL	
Description	Description of the auction item	Text	-	Yes	NULL	
Start_Time	Auction start time	DateTime	-	No	NULL	
End_Time	Auction end time	DateTime	-	No	NULL	
Starting_Bid	Minimum starting bid	Float	-	No	NULL	
Current_High_Bid	Current highest bid	Float	-	Yes	NULL	
User_ID	ID of user creating	Integer	-	No	NULL	FK (User.I

Column Name	Description	Type	Length	Nullable	Default Value	Key Type
	the auction					D)
Created_At	Auction creation timestamp	DateTime	-	No	CURRENT_TIMESTAMP	

7.2.4 Data 4: Bid

Name Bid

Alias Offer

Where-used/how-used Used for placing and tracking bids in auctions.

Content Description ID + Amount + Timestamp + User_ID + Auction_ID

Table: Bid

Column Name	Description	Type	Length	Nullable	Default Value	Key Type
ID	Unique identifier for the bid	Integer	-	No	Auto-increment	PK
Amount	Bid amount placed by the user	Float	-	No	NULL	
Timestamp	Time the bid was placed	DateTime	-	No	CURRENT_TIMESTAMP	
User_ID	ID of user placing the bid	Integer	-	No	NULL	FK (User.ID)
Auction_ID	ID of auction the bid belongs to	Integer	-	No	NULL	FK (Auction.ID)

Data 5: Notification

Name Notification

Alias Alert

Where-used/how-used Used to notify users about auction and bid updates.

Content Description ID + Message + User_ID + Timestamp

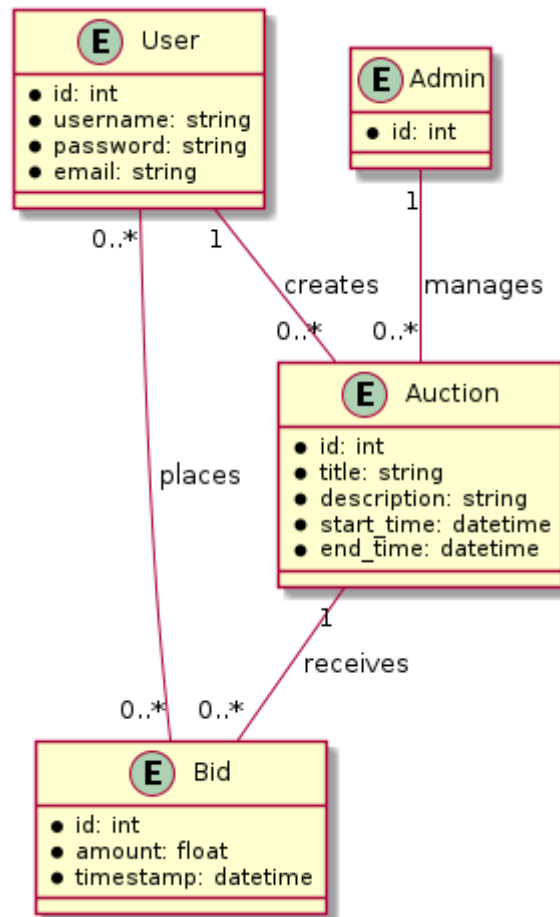
Name Notification

Table: Notification

Column Name	Description	Type	Length	Nullable	Default Value	Key Type
ID	Unique identifier for notification	Integer	-	No	Auto-increment	PK
Message	Notification content	Text	-	No	NULL	
User_ID	ID of user receiving the notification	Integer	-	No	NULL	FK (User.ID)
Timestamp	Time the notification was sent	DateTime	-	No	CURRENT_TIMESTAMP	

ER

DIAGRAM



7.2.1.2 ER Data Model

1. User Table

Attribute Name	Data Type	Constraints
ID	Integer	Primary Key, Not null, Unique
Username	String	Not null
Password	String	Not null
Email	String	Not null, Unique

2. Admin Table

Attribute Name	Data Type	Constraints
ID	Integer	Primary Key, Not null, Unique

4. Auction Table

Attribute Name	Data Type	Constraints
ID	Integer	Primary Key, Not null, Unique
Title	String	Not null
Description	String	Optional
Start_Time	DateTime	Not null
End_Time	DateTime	Not null

5. Bid Table

Attribute Name	Data Type	Constraints
ID	Integer	Primary Key, Not null, Unique
Amount	Float	Not null
Timestamp	DateTime	Not null
User_ID	Integer	Foreign Key references User(ID)
Auction_ID	Integer	Foreign Key references Auction(ID)

5 Relationships

Relationship	Source Table	Source Attribute	Target Table	Target Attribute	Cardinality
User places Bid	User	ID	Bid	User_ID	1 to Many
Bid is placed on Auction	Bid	Auction_ID	Auction	ID	Many to 1
Admin creates Auction	Admin	ID	Auction	Admin_ID	1 to Many
Admin manages Auction	Admin	ID	Auction	Admin_ID	Many to Many*

7.2.1.3 E/R Model Description

1. Entities:

- **User:** Represents the users participating in the auction system.
- **Admin:** Represents the administrators managing the auctions.
- **Auction:** Represents the auctioned items.
- **Bid:** Represents the bids placed by users on auctions.

2. Relationships:

- **User places Bid:** A user can place multiple bids, but each bid is associated with one user.
- **Bid is placed on Auction:** A bid is linked to one auction, and an auction can have multiple bids.
- **Admin creates Auction:** Admins are responsible for creating auctions.
- **Admin manages Auction:** Admins also manage the auctions after they are created.

3. Cardinalities:

- A **User** can place **0 or more Bids**, but a **Bid** is placed by **1 User**.
- A **Bid** belongs to **1 Auction**, but an **Auction** can receive **0 or more Bids**.
- An **Admin** can create or manage **0 or more Auctions**.

7.2.2 Data Dictionary

7.2.2.1 Data 1: User

Attribute Name: User ID

Description: Unique identifier for each user.

Data Type: Integer

Primary Key: Yes

Constraints: Not null, Unique

Attribute Name: User Name

Description: Name of the user.

Data Type: String

Constraints: Not null

Attribute Name: Email

Description: Email address of the user.

Data Type: String

Constraints: Not null, Unique

Attribute Name: Password

Description: Encrypted password for the user.

Data Type: String

Constraints: Not null
Attribute Name: Contact Number

Description: Phone number of the user.
Data Type: String
Constraints: Optional

7.2.2.2 Data 2: Item

Attribute Name: Item ID

Description: Unique identifier for each auction item.
Data Type: Integer
Primary Key: Yes
Constraints: Not null, Unique
Attribute Name: Item Name

Description: Name of the auction item.
Data Type: String
Constraints: Not null
Attribute Name: Item Description

Description: Detailed description of the auction item.
Data Type: Text
Constraints: Optional
Attribute Name: Starting Bid

Description: Minimum bid amount for the item.
Data Type: Decimal
Constraints: Not null
Attribute Name: Auction End Time

Description: End time of the auction for the item.
Data Type: Date Time
Constraints: Not null

7.2.2.3 Data 3: Bid

Attribute Name: Bid ID

Description: Unique identifier for each bid.
Data Type: Integer
Primary Key: Yes

Constraints: Not null, Unique
Attribute Name: Bid Amount

Description: Amount placed by the bidder.
Data Type: Decimal
Constraints: Not null
Attribute Name: Bid Time

Description: Time when the bid was placed.
Data Type: Date Time
Constraints: Not null
Attribute Name: User ID

Description: Identifier of the user who placed the bid.
Data Type: Integer
Constraints: Foreign Key (references User ID in User table)
Attribute Name: Item ID

Description: Identifier of the item being bid on.
Data Type: Integer
Constraints: Foreign Key (references Item ID in Item table)

7.2.2.4 Data 4: Chat

Attribute Name: Chat ID

Description: Unique identifier for each chat instance.
Data Type: Integer
Primary Key: Yes
Constraints: Not null, Unique
Attribute Name: Sender ID

Description: Identifier of the user sending the message.
Data Type: Integer
Constraints: Foreign Key (references User ID in User table)
Attribute Name: Receiver ID

Description: Identifier of the user receiving the message.
Data Type: Integer
Constraints: Foreign Key (references User ID in User table)
Attribute Name: Message Content

Description: Content of the chat message.
Data Type: Text
Constraints: Not null
Attribute Name: Timestamp

Description: Time when the message was sent.

Data Type: Date Time
Constraints: Not null

7.2.2.5 Data 5: Notification

Attribute Name: Notification ID

Description: Unique identifier for each notification.

Data Type: Integer

Primary Key: Yes

Constraints: Not null, Unique

Attribute Name: Notification Type

Description: Type of notification (e.g., Bid Update, Auction Status).

Data Type: String

Constraints: Not null

Attribute Name: User ID

Description: Identifier of the user receiving the notification.

Data Type: Integer

Constraints: Foreign Key (references User ID in User table)

Attribute Name: Message Content

Description: Details of the notification.

Data Type: Text

Constraints: Optional

Attribute Name: Timestamp

Description: Time when the notification was sent.

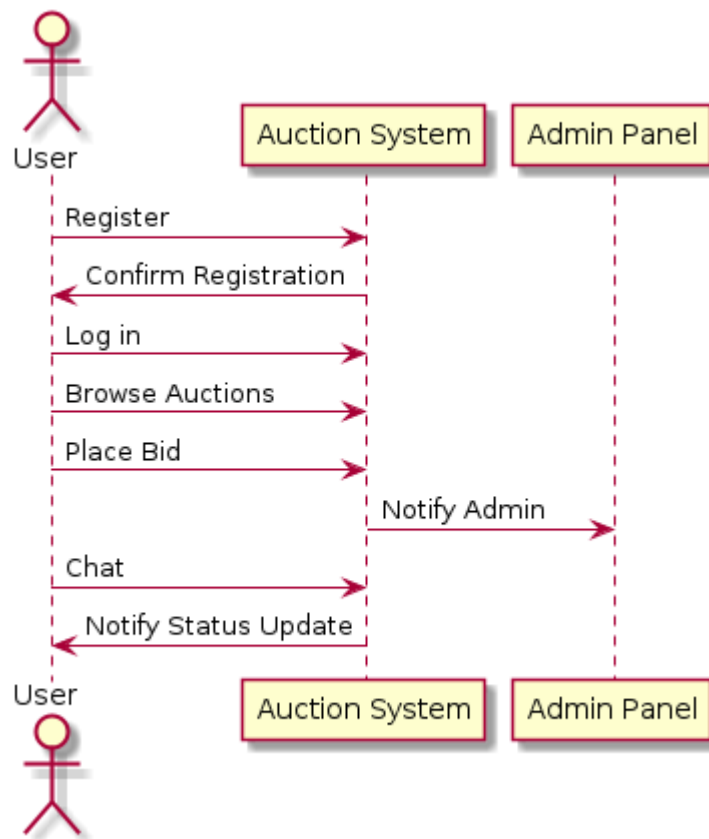
Data Type: Date Time

Constraints: Not null

Application Design

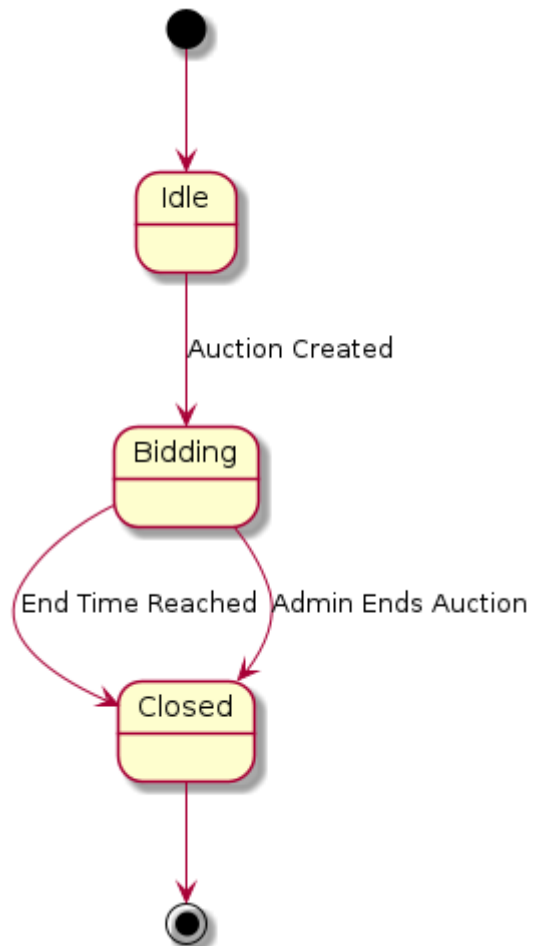
7.4.1 Sequence Diagram

Sequence Diagram 1



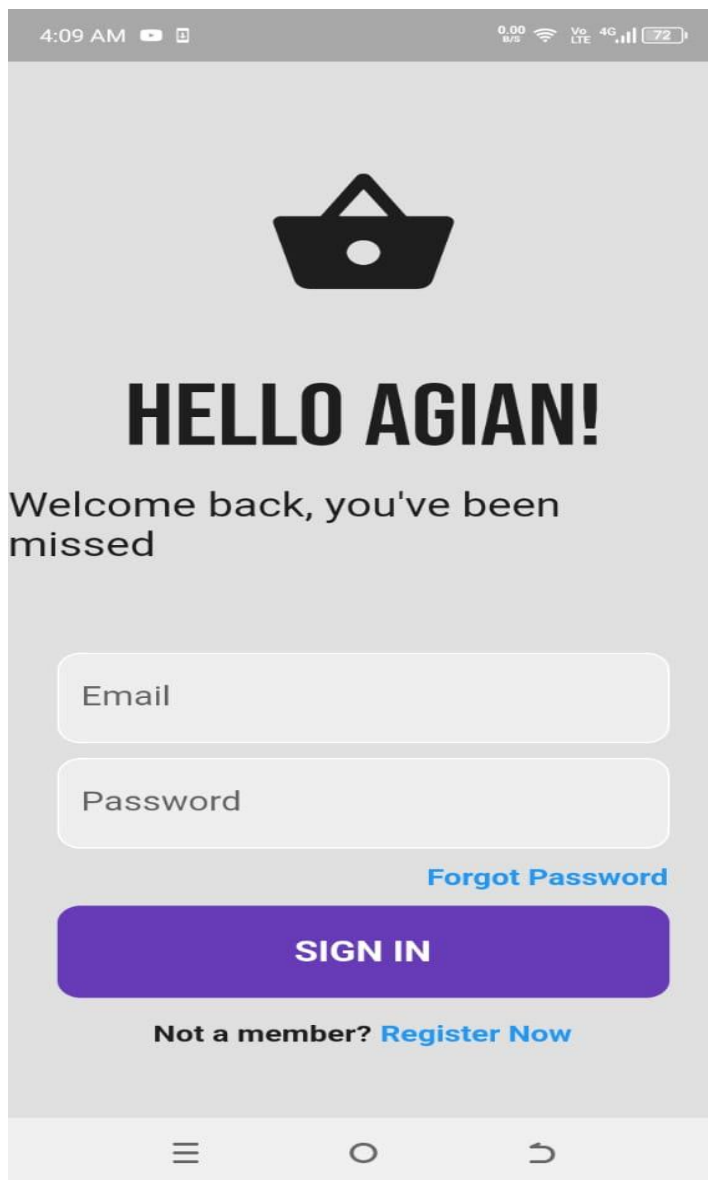
State Diagram

State Diagram 1



GUI Design

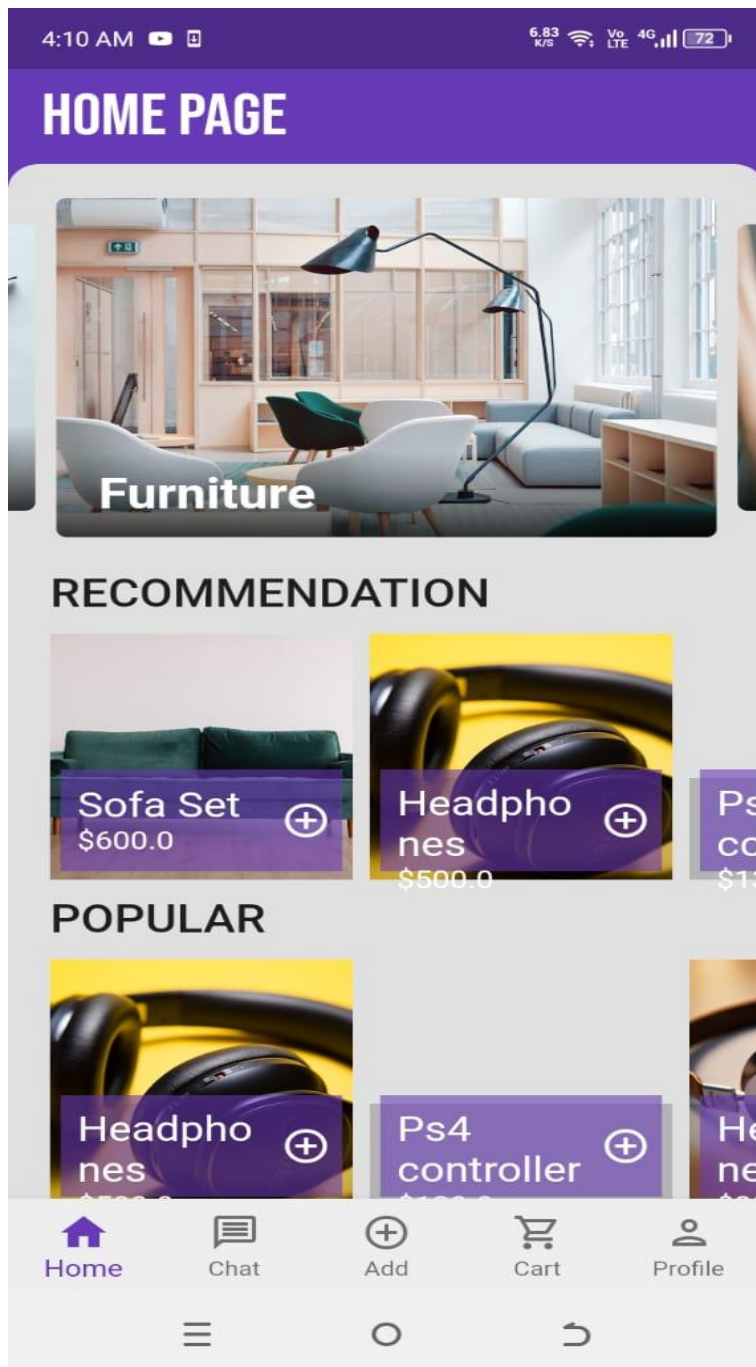
7.5.1 Login Page - Mock Screen 1



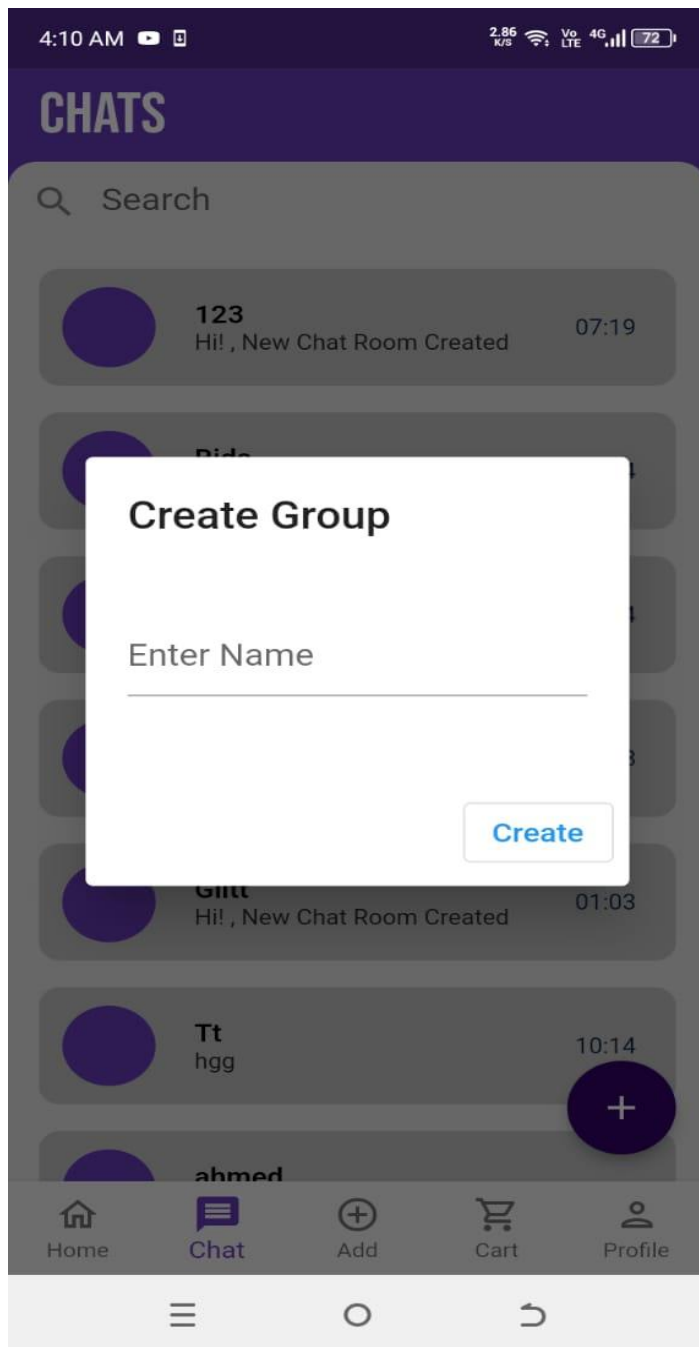
7.5.2 Sign Up Page - Mock Screen 2

The image shows a mobile application mockup for a registration screen. At the top, the status bar displays the time as 4:09 AM, along with icons for YouTube, a mobile app, data speed (0.00 B/s), VoLTE, 4G signal, and a battery level of 72%. The main heading is "HELLO THERE" in large, bold, black letters. Below it, the text "Register below with your details!" is displayed. The registration form consists of five input fields: "Username", "Contact" (with a character count of 0/11), "Email", "Password", and "Confirm Password". All fields are currently empty. Below the form is a prominent purple button labeled "SIGN UP". At the bottom of the form area, there is a link that says "I am a member! [Login Now](#)". The bottom of the screen features a standard Android navigation bar with three icons: a hamburger menu, a home circle, and a back arrow.

7.5.3 HomePage - Mock Screen 3



7.5.4 Chat Room - Mock Screen 4



7.5.5 Launch Product - Mock Screen 5

4:10 AM 1.94 K/S Vo LTE 4G 72

ADD PRODUCT

FILL TO LAUNCH

Product Name

Select Product Category

Image URL

Starting Bid

End of Session

Product Details

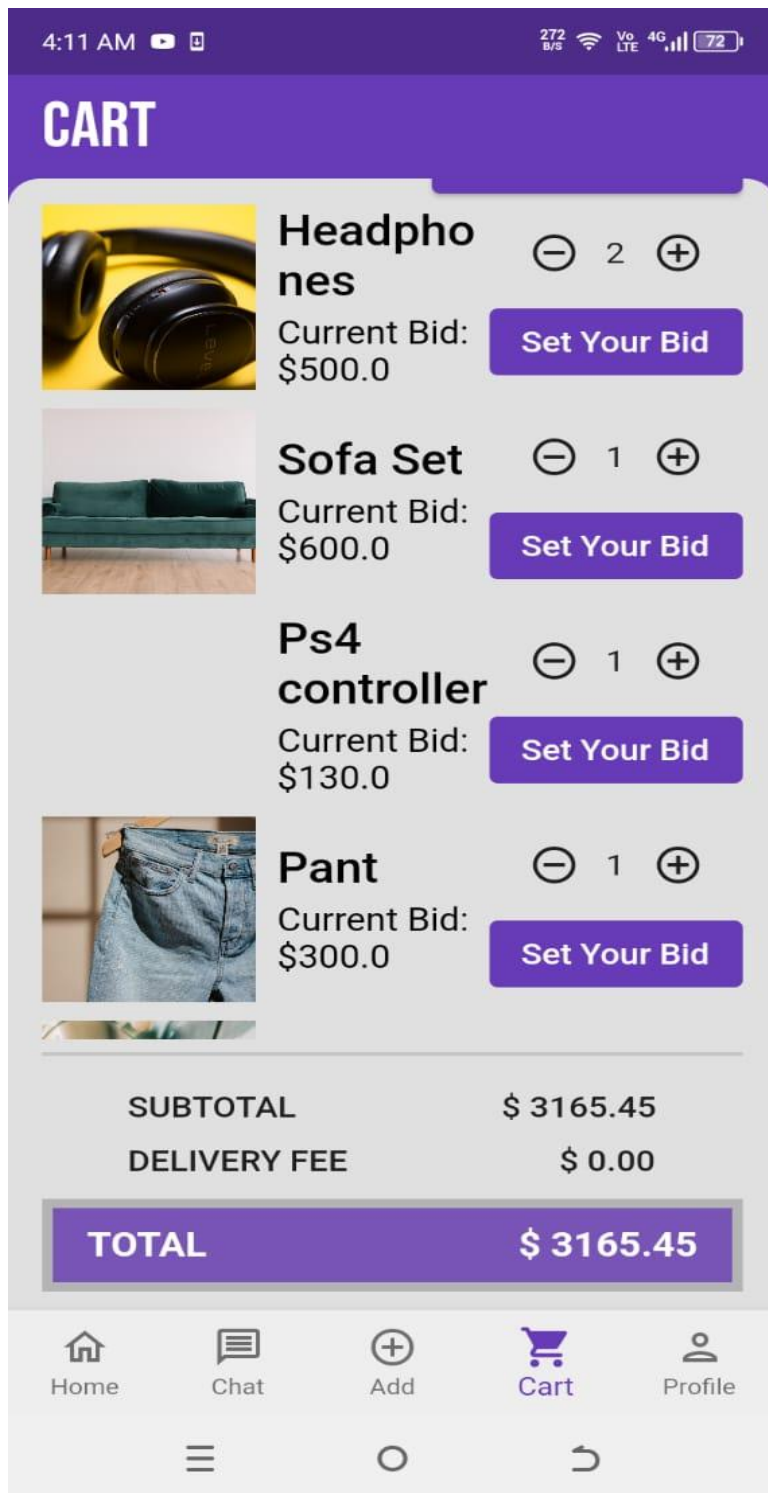
Delivery Details

Launch Product

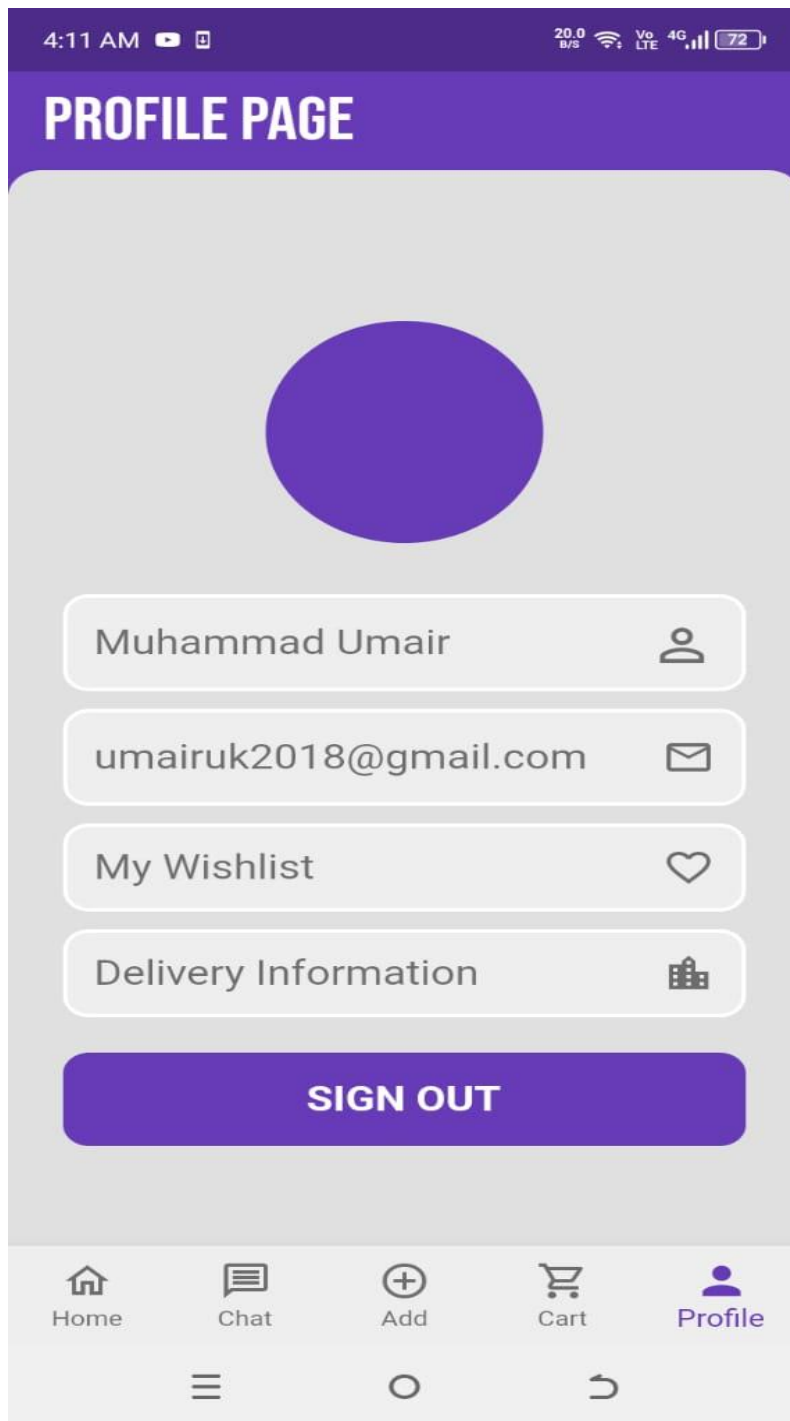
Home Chat **Add** Cart Profile

≡ ○ ↺

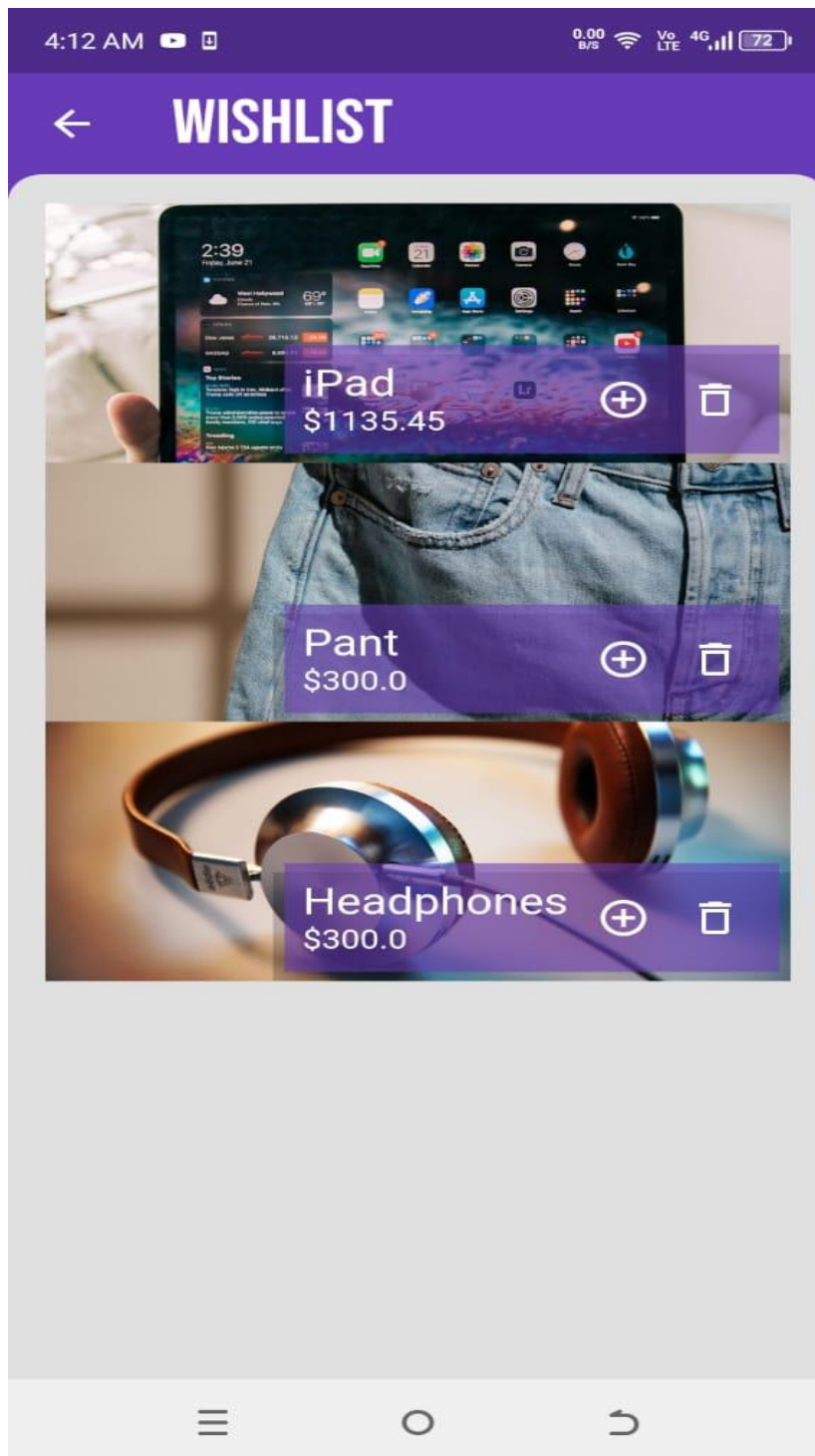
7.5.6 Add To Cart - Mock Screen 6



7.5.7 Profile Page - Mock Screen 7



7.5.8 WhishList - Mock Screen 8



7.5.9 Delivery Information - Mock Screen 9

4:12 AM 0.00 B/s Vo LTE 4G 72

← **DELIVERY INFORMATION**

FILL TO START ORDERING

Address

City

Country

ZIP Code

Save Information

≡ ○ ↷

References

D. Gupta, "How to Build an Online Auction Application? Benefits, Features, Costs", 2024.

G. Patil, "Online Auction App Development: A Complete Guide", India (Delhi), 2022.

J. Tiwary, "Auction App Development: Unlocking the Future of Online Bidding", India (Mumbai), 30 October, 2023.

M. Butt, "How to Create an Online Real-Time Bidding/Auction App", Mumbai (bendra), 2022.

Ajay, "How to Develop an Application for Online Auctions? Features, Advantages, and Costs", Chennai, 2024.

A4. OTHER TECHNICAL DETAIL DOCUMENTS

Test Cases Document

S. No	Description	Test Engineer	Start Date	End Date
1	Validate user login with correct and wrong info	Bismah	02-Feb-2025	03-Feb-2025
2	Check user registration and email verification	Umair	04-Feb-2025	05-Feb-2025
3	Test homepage tiles: Auction list, bids, chat etc.	Maham, Bismah	06-Feb-2025	07-Feb-2025
4	Post a product for auction	Bismah	10-Feb-2025	11-Feb-2025
5	Place a bid and test real-time update	Maham,Umair	12-Feb-2025	13-Feb-2025
6	Use chat room to message a seller	Bismah , Maham	14-Feb-2025	15-Feb-2025
7	Edit user profile and update password	Maham	17-Feb-2025	18-Feb-2025
8	Fill delivery info screen and submit	Umair	19-Feb-2025	20-Feb-2025
9	Add/remove item from wishlist	Bismah	21-Feb-2025	22-Feb-2025
10	View auction report with bidding history	Maham	10-May-2025	10-May-2025
11	View user activity report	Umair,Maham	11-May-2025	11-May-2025
12	Check auction summary report	Maham	12-May-2025	12-May-2025
13	View report of completed payments	Maham	13-May-2025	13-May-2025

Module Name: Login Screen

Date: 02-Feb-2025

Test Case ID: TC-LS-001

Test Engineer: Bismah

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Enter correct email and password	Bismahimran31@gmail.com / 1234	Login successful, goes to homepage	As expected	Pass
2	Enter correct email but wrong password	Bismahimran31@gmail.com / 0000	Show "Invalid login" error	As expected	Pass
3	Leave password field empty	bismah@gmail.com / -	Show "Password is required"	As expected	Pass
4	Click on "Forgot Password"		Redirect to reset screen	As expected	Pass

Module Name: Registration Screen

Date: 04-Feb-2025

Test Case ID: TC-RS-001

Test Engineer: Umair

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Fill name, email, password	Maham / maham@gmail.com / xyz123	Registration successful	As expected	Pass
2	Submit with	-	Show "All fields	As	Pass

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
	empty fields		required"	expected	
3	Enter invalid email format	abc@com	Show "Enter valid email"	As expected	Pass

Module Name: Bidding Screen

Date: 6-Feb-2025

Test Case ID: TC-BS-001

Test Engineer: Maham, Bismah

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	View product auction	-	Show item details with bid option	As expected	Pass
2	Place higher bid	1500	Bid accepted, status updated	As expected	Pass
3	Place lower bid	1200	Show "Bid too low" message	As expected	Pass

Module Name: Product Posting Screen

Date: 10-Feb-2025

Test Case ID: TC-PRD-004

Test Engineer: Maham Mirza

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Open "Post Product" screen	-	Form with fields is displayed	As expected	Pass
2	Enter item title and description	"Branded Shoes", "Limited Ed"	Title and description accepted	As expected	Pass
3	Upload image	image.jpg	Image uploaded and previewed	As expected	Pass
4	Submit without starting bid	-	Show error: "Starting bid required"	As expected	Pass

Module Name: Place Bid Screen

Date: 12-Feb-2025

Test Case ID: TC-BID-005

Test Engineer: Maham Mirza

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Select product	Product ID	Product detail page is shown	As expected	Pass
2	Place higher bid	1500	Bid placed successfully	As expected	Pass
3	Place lower bid	900	Error: Bid must be higher than current	As expected	Pass

Module Name: Chat Room

Date: 14-Feb-2025

Test Case ID: TC-CR-001

Test Engineer: Bismah, Maham

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Open chat screen	-	Chat loads with messages	As expected	Pass
2	Send message to seller	"Is it available?"	Message delivered to seller	As expected	Pass
3	Receive reply	-	New reply message shown in chat	As expected	Pass

Module Name: User profile and Update pass

Date: 17-Feb-2025

Test Case ID: TC-UP-001

Test Engineer: Maham Mirza

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Open Profile screen	-	Display user details (name, email etc.)	As expected	Pass
2	Tap on Edit Profile	-	Editable fields appear	As expected	Pass
3	Update name and phone number	Name: "Bismah", Phone: "0300..."	Changes saved and updated on profile	As expected	Pass
4	Leave required field empty	Name: ""	Show error: "Name is required"	As expected	Pass
5	Change password	Old: 1234, New: xyz@2025	Password updated with confirmation	As expected	Pass

Module Name: Delivery Info Screen

Date: 19-Feb-2025
Test Case ID: TC-DEL-008
Test Engineer: Umair

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Fill delivery form	Name, Address, Phone	Details accepted and saved	As expected	Pass
2	Submit with empty fields	-	Show error: "All fields required"	As expected	Pass
3	View confirmation message	-	"Delivery info submitted" shown	As expected	Pass

Module Name: Wishlist Screen

Date: 21-Feb-2025
Test Case ID: TC-WS-001
Test Engineer: Bismah

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Add item to wishlist	Tap on "♡" icon	Item added to wishlist	As expected	Pass
2	Remove item from wishlist	Tap on "♡" again	Item removed from wishlist	As expected	Pass

Module Name: Bidding History

Date: 10-May-2025

Test Case ID: TC-RPT-001

Test Engineer: Maham

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Open Bidding History Report	-	Shows list of bids placed	As expected	Pass
2	Click on specific auction	-	Shows all bids with user names	As expected	Pass

Module Name: User Activity

Date: 11-May-2025

Test Case ID: TC-RPT-011

Test Engineer: Umair,Maham

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Open User Activity Report	-	Shows user login, posting, and bidding history	As expected	Pass
2	Select user	User ID	Displays that user's full activity timeline	As expected	Pass

Module Name: Auction Summary

Date: 12-May-2025

Test Case ID: TC-RPT-012

Test Engineer: Maham

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Open Auction Summary	-	List of all auctions with summary shown	As expected	Pass
2	Click specific auction	Auction ID	Shows full details of selected auction	As expected	Pass
3	View confirmation message	-	"Delivery info submitted" shown	As expected	Pass

Module Name: Payment Status

Date: 13-May-2025

Test Case ID: TC-RPT-013

Test Engineer: Maham

S. No	Steps	Input Data	Expected Result	Actual Result	Pass/Fail
1	Open Payment Status Report	-	Displays list of completed payments	As expected	Pass
2	View payment for one auction	Auction ID	Shows buyer name and payment status	As expected	Pass

UI/UX Detail Document

1. Introduction

This document explains how the design of the Online Auction App makes it easy and enjoyable to use. UI (User Interface) is about what users see, like buttons and screens, while UX (User Experience) is about how users feel when using the app — whether it's smooth, fast, and frustration-free.

Our goal is to build an app that is simple, clear, and works well on any mobile phone, so that users can buy, sell, and bid on items without confusion.

2. Design Principles

We followed these **easy-to-understand rules** to make sure the app is friendly and looks good:

- **Simplicity**
 - The app shows **only what is necessary**, so users don't feel overwhelmed.
 - Clean layout with readable fonts and good spacing.
- **Consistency**
 - Same **color theme, button styles, and icons** throughout the app.
- **Mobile First**
 - The design is made **especially for phones**, but it also works on tablets.
- **Feedback and Response**
 - When users take an action (like placing a bid), the app instantly shows a result (success message, or error).
- **Accessibility**
 - Easy to use for **all types of users**, including those who may have difficulty seeing or reading.

4. UI Components

Page	What It Does
Login Screen	Lets users enter their email and password to sign in.
Sign Up Screen	New users can create an account.
Homepage	Shows ongoing auctions, popular items, and buttons for navigation.
Auction Detail Page	Shows item picture, description, current bid, and bid button.

Page	What It Does
Post Product Screen	Sellers enter item details and upload an image.
Chat Room	Allows buyers and sellers to message each other.
Wishlist	Users can save favorite items.
Delivery Info Page	Buyers enter address and phone number for delivery.
Profile Page	View and update user info like name and password.
Notification Popups	Show when bids are placed or auctions end.

5. User Flow

- Buyer's Flow

1. Open app → Login or Sign up
2. Browse products → Tap on an auction
3. Place a bid → Get updates
4. Win auction → Enter delivery info
5. Receive item

- Seller's Flow

1. Login → Tap "Post Product"
2. Add item details → Set auction time
3. Submit → View bids in real-time
4. Message buyer → Deliver item

- Admin's Flow

1. Login → Manage users and items
2. View reports → Block spam users
3. Monitor activity

5. Tools and Technologies Used

Tool	Purpose
Flutter	A tool used to design beautiful mobile apps (works for Android & iOS).
Firebase	A Google tool that stores user data, sends notifications, and handles login securely.
Dart	The programming language used to write the app in Flutter.
Figma/Canva (for Mockups)	Used to create the screen designs before building the real app.
GitHub	Stores our code safely and allows teamwork.

Summary

The UI/UX of our Online Auction App is carefully designed to be:

- Easy for first-time users
- Fast and mobile-friendly
- Visually clean and consistent
- Helpful through instant responses and notifications

Coding Standards Document

1. Introduction

This document explains the **coding rules** and **good habits** our team followed while building the Online Auction App. Following these standards helps us keep the code:

- Clean and organized
- Easy to understand
- Secure
- Easy to update later

2. General Guidelines

These are some basic rules that apply to all parts of the app:

- **Use meaningful names**
Example: placeBid is better than pb because it tells you what it does.
- **Keep your code neat**
Indent properly, leave spaces where needed, and avoid writing everything in one line.
- **Write comments**
Add short notes in the code to explain tricky parts.
- **Avoid repeating code**
Use functions to avoid writing the same code again and again.
- **Use version control**
All changes were saved using GitHub so nothing is lost and we can go back to older versions if needed.

3. Frontend Coding Standards (Flutter with Dart)

- **File and Folder Structure**
 - Each screen has its own folder (e.g., login/, auction/)
 - All images stored in assets/images/
- **Naming Rules**
 - Files: lowercase_with_underscores.dart (e.g., login_screen.dart)
 - Classes: PascalCase (e.g., LoginScreen)

- Variables/Functions: camelCase (e.g., userName, placeBid())
- **Widget Structure**
 - Use StatelessWidget when UI doesn't change.
 - Use StatefulWidget when UI updates (like when bidding).
- **UI Testing**
 - Used flutter_test to test if widgets appear correctly (e.g., checking if login button shows up).

4. Backend Coding Standards

The backend logic is handled using **Firestore** with **Dart services**.

- **API Integration**
 - Used Firestore APIs for authentication, database (Firestore), and messaging.
- **Code Organization**
 - auth_service.dart for login/signup
 - auction_service.dart for managing auctions
 - chat_service.dart for handling messages
- **Function Standards**
 - Small and focused: Each function does one thing.
 - Clear error messages: Show helpful alerts if something goes wrong.
- **Backend Testing**
 - Checked if data is correctly saved, like:
 - Can user sign in?
 - Is a bid added to the database?
 - Can admin remove a user?

5. Database Standards

- **Collection Naming**
 - Collection names are plural and lowercase: users, auctions, bids
- **Timestamps**
 - Always saved in standard format: yyyy-mm-dd hh:mm

6. Security Practices

We followed important safety rules to keep user data and transactions secure.

- **Passwords & Login**
 - Passwords are never stored as plain text.
 - Used Firebase Authentication for secure login.
 - Email verification is required.
- **Data Protection**
 - Data transferred between users and app is encrypted.
 - Only logged-in users can access their data.
- **Role-Based Access**
 - Users can only do what their role allows:
 - Buyers can't delete auctions.
 - Admins have full control.
- **Common Risks Avoided**
 - Prevented fake bids using real-time validation.
 - Blocked invalid login attempts with error alerts.
 - Monitored user activity to detect spam or abuse.

Summary

By following these coding standards, we:

- Keep the code neat and easy to understand
- Make the app secure and stable
- Allow future team members to maintain or improve the app easily

Project Policy Document

1. Introduction

This document explains the **rules, responsibilities, and working method** of our Online Auction App project. It helps all team members and supervisors understand how we work, what everyone's role is, and how we keep the project on track.

Our goal is to **build a secure, easy-to-use app** where people can buy and sell items through bidding. To do that smoothly, we need good teamwork, communication, and clear policies.

2. Getting Started

This section helps understand how the project begins and how we organize our tasks.

- **Team Members & Roles**

Name	Role
M. Umair	Team Leader (Manages tasks, reviews work)
Bismah Imran	Developer & Tester (Designs and tests screens)
Maham Mirza	Developer & Tester (Builds features, reports bugs)
Osama Ahmed Khan	Supervisor (Guides the team and approves work)

- **Timeline / Working Plan**

Phase	Time Period
Planning & Research	Noc 2024
Designing Screens	Jan 2025
Development (Coding)	Feb – Mar 2025
Testing & Fixing Bugs	April 2025
Final Submission	June 2025

- **Tools We Use**

Tool	Why We Use It
Flutter	To build the app interface
Firebase	To save user data and manage login
GitHub	To store code and track changes
WhatsApp	For team communication
Google Docs	For writing and sharing reports

3. Features of Our Working Style

- **Team Rules**

- Everyone must attend weekly meetings.
- Everyone should complete their assigned tasks on time.
- Help each other and respect everyone's ideas.

- **Coding & Designing Rules**

- Follow neat and clean coding.
- Design screens that are easy to use and look good.
- Test everything before submitting.

- **Communication**

- Use WhatsApp for quick updates.
- Share progress in group meetings.
- Inform the team in case of any delay or problem.

- **Safety & Backup**

- Save project files on GitHub to avoid losing work.
- Backup Firebase data weekly.

- **Testing & Feedback**

- Every feature must be tested before final use.
- Bugs are fixed during the testing phase.
- Users should get easy error messages if something goes wrong.

- **Fair Use & Ethics**

- No copying others' code or using illegal software.
- Keep all user data private and protected.
- Only real items should be posted for auction no scams.

User Manual Document

1. Introduction

Welcome to the Online Auction App is a simple and fun way to buy and sell items through bidding. This app lets you find great deals, post your own items for sale, and take part in auctions in real time all from your mobile phone.

2. Key Features

Feature	What It Does
Register/Login	Create an account or log in securely.
Browse Items	See different items up for auction.
Place Bids	Take part in auctions and offer your price.
Post Your Item	Sell your item by setting a starting bid and auction time.
Chat	Talk to buyers or sellers directly inside the app.
Notifications	Get instant alerts for new bids or auction results.
Wishlist	Save items you like for later.
Delivery Info	Enter your address and phone number after winning.
Profile Settings	Change your personal info or password anytime.

3. Getting Started

Step 1: Install the App

- Download from the Google Play Store (Android only).
- Open the app on your mobile phone.

Step 2: Create an Account

- Tap on Sign Up.
- Fill in your name, email, password.
- You'll get a verification email — open it and click the link.

Step 3: Log In

- Use your email and password to log in.
- You'll be taken to the homepage where you can start browsing.

Step 4: Start Using the App

- **Buyers:** Tap on an item then Place bid then Get alerts.
- **Sellers:** Tap Post Item then Add title, image, price then Start auction.

4. Troubleshooting

Problem	What to Do
Can't log in	Check your email/password. Use "Forgot Password" if needed.
Didn't get email verification	Check your Spam or Junk folder. Tap "Resend" in the app.
App is slow or stuck	Close and reopen the app. Make sure your internet is working.
Can't place a bid	Make sure your bid is higher than the current one.
Chat not working	Try refreshing the app or check your internet connection.
Delivery info not saving	Make sure all fields (name, phone, address) are filled correctly.

5. Support

If you still need help, here's how you can contact us:

- **Email Support:** support@auctionapp.com
- **In-App Help:** Go to Profile → Help Center
- **Live Chat:** Available 10:00 AM – 6:00 PM (Mon–Sat)
- **FAQs:** Check the section in the app.

A11. Plagiarism Test Summary Report