ONLINE CLASSIFIED MARKETPLACE

A DESIGN PROJECT REPORT

submitted by

SRIHARI PRASAD S

VINOTH T

VIJAY V

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

K RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai, Approved by AICTE, New Delhi)

Samayapuram – 621 112

DECEMBER, 2024

ONLINE CLASSIFIED MARKETPLACE

A DESIGN PROJECT REPORT

submitted by

SRIHARI PRASAD S (811722104153)

VIJAY V (811722104182)

VINOTH T (811722104185)

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

K RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai, Approved by AICTE, New Delhi)

Samayapuram – 621 112

DECEMBER, 2024

K RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report titled **"ONLINE CLASSIFIED MARKETPLACE"** is bonafide work of **SRIHARI PRASAD S (811722104153), VIJAY V (811722104182), VINOTH T (811722104185)** who carried out the project under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE
DrA Delphin Carolina Rani M.E.,Ph.D.,
HEAD OF THE DEPARTMENT
PROFESSOR
Department of CSE
K Ramakrishnan College of Technology
(Autonomous)
Samayapuram – 621 112

SIGNATURE
Ms. P. Karthika, M.E.,
SUPERVISOR
Assistant Professor
Department of CSE
K Ramakrishnan College of Technology
(Autonomous)
Samayapuram – 621 112

Submitted for the viva-voice examination held on ……

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

We jointly declare that the project report on **"ONLINE CLASSIFIED MARKETPLACE"** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of Bachelor Of Engineering. This project report is submitted on the partial fulfilment of the requirement of the awardof Degree of Bachelor Of Engineering.

Signature

SRIHARI PRASAD S

VINOTH T

VIJAY V

Place: Samayapuram

Date:

ACKNOWLEDGEMENT

It is with great pride that we express our gratitude and indebtness to our institution **"K RAMAKRISHNAN COLLEGE OF TECHNOLOGY"**, for providing us with the opportunity to do this project.

We are glad to credit honorable chairman **Dr. K RAMAKRISHNAN, B.E.,** for having provided for the facilities during the course of our study in college.

We would like to express our sincere thanks to our beloved Executive Director **Dr. S KUPPUSAMY, MBA, Ph.D.,** for forwarding our project and offering adequate duration to complete it.

We would like to thank **Dr. N VASUDEVAN, M.Tech., Ph.D.,** Principal, who gave opportunity to frame the project with full satisfaction.

We whole heartily thank **Dr. A DELPHIN CAROLINA RANI, M.E., Ph.D.,** Head of the Department, **COMPUTER SCIENCE AND ENGINEERING** for providing her support to pursue this project.

We express our deep and sincere gratitude and thanks to our project guide **Ms. P KARTHIKA, M.E.,** Department of **COMPUTER SCIENCE AND ENGINEERING,** for her incalculable suggestions, creativity, assistance and patience which motivated us to carry our this project.

We render our sincere thanks to Course Coordinator and other staff members for providing valuable information during the course. We wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

ABSTRACT

This project introduces a versatile online platform that facilitates secure and efficient peer-to-peer buying and selling of products, with integrated real-time chat and admin-mediated transactions. Leveraging Firebase for robust backend support, the platform offers comprehensive user profile management, authentication, product uploads, and advanced search functionality, making product discovery simple and intuitive. Buyers and sellers can communicate directly through live chat to discuss product details, negotiate terms, and complete transactions with added security. The platform ensures customer protection with UPI-based payments, delivery confirmations, and admin oversight to verify transactions. A user-friendly settings page allows for profile updates, feedback submission, and quick access to contact and support options, ensuring an intuitive navigation experience. With added features for reviewing, rating, and reporting, the platform promotes a trustworthy and interactive marketplace. This project aims to enhance digital commerce accessibility, improve user experience, and foster safe interactions in online peer-to-peer transactions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATION	FULL FORM
API	Application Programming Interface
CLI	Command Line Interface
SSL	Secure Sockets Layer
RTDB	Real-Time Database
AI	Artificial Intelligence
AR	Augmented Reality
UPI	Unified Payment Interface
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheet
SQL	Structured Query Language
UID	User Identifier
JS	JavaScript

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

The evolution of e-commerce has transformed how individuals buy and sell products, making online marketplaces increasingly popular. However, ensuring secure, user-friendly, and trustworthy transactions in peer-to-peer platforms remains a challenge. This project addresses these needs by creating a comprehensive online platform with a focus on secure transactions and interactive buyer-seller communication. Leveraging Firebase as the backend, the platform integrates user authentication, product management, and real-time chat for efficient buyer-seller interactions. A unique feature is the inclusion of an admin-mediated transaction system using UPI, which safeguards each transaction by requiring delivery verification and allowing buyer-seller rating and feedback. Inspired by the principles of user-centered design, the platform also includes an intuitive settings page for profile management, feedback submission, and customer support. Recent advancements in mobile payment systems, cloud computing, and real-time databases have enabled the development of reliable, scalable, and secure online platforms. This project combines these innovations to create a safe, accessible, and engaging digital marketplace.

Figure 1.1: Flow of Online Classified Marketplace

1.2 OVERVIEW

This project introduces a comprehensive and secure peer-to-peer (P2P) platform tailored for buying and selling products, bringing a new level of interactivity, trust, and convenience to online transactions. The platform leverages Firebase for a robust backend infrastructure, incorporating user authentication, secure transactions, and real-time communication between buyers and sellers. One of the unique features of this system is an admin-mediated transaction model, which provides an additional layer of security for both parties by overseeing payments and delivery confirmations. This ensures that transactions are conducted smoothly and securely, utilizing UPI for payment processing and verification, making it particularly suitable for markets where digital payment options are common.Users can create profiles, upload products with detailed descriptions, and search or filter products based on various attributes, such as name, category, or user ratings. Product discovery is made intuitive through an advanced search feature, allowing buyers to quickly find relevant listings. Real-time chat functionality fosters direct communication between buyers and sellers, making it easy to discuss product details and negotiate terms, while feedback and rating options enable users to establish a reputation within the marketplace. In addition, the settings page provides users with profile management capabilities, feedback submission forms, and easy access to support options, ensuring a user-friendly experience. The platform also emphasizes user accountability through review and reporting features, allowing users to report inappropriate listings or experiences, which the admin team can review.Drawing from recent advancements in cloud-based solutions, mobile payment technology, and real-time database systems, this platform combines functionality, security, and ease of use to create an accessible and trustworthy digital marketplace. By integrating essential features with user-centered design principles, this project provides a safe, efficient, and interactive space for online peer-to-peer transactions, appealing to users seeking a reliable and user-friendly online shopping experience.

1.3 IMPLICATION

This project is a cutting-edge peer-to-peer marketplace that prioritizes security, user convenience, and transparency. It leverages Firebase as a robust backend solution to manage authentication, real-time communication, data storage, and secure transactions. The platform is designed to facilitate seamless interactions between buyers and sellers while ensuring accountability through a structured system of checks and balances.

A key feature is its **real-time chat** functionality, allowing users to communicate directly. Buyers and sellers can discuss products, negotiate deals, and finalize transactions efficiently. The chat system integrates status indicators to ensure users are aware of the availability of their counterparts, fostering real-time responsiveness.To ensure safe transactions, the system adopts an **admin-mediated payment process**. Buyers transfer funds through a secure payment gateway, with the admin acting as an intermediary. This process guarantees that payments are released to sellers only after delivery confirmation, safeguarding both parties against fraud.The platform supports product uploads and searches, enabling sellers to list items under defined categories with detailed descriptions and images. Buyers can filter and locate products effortlessly, whether by name, description, or category-specific buttons, creating an intuitive shopping experience.

User feedback is another pillar of the system. Buyers and sellers can leave reviews and ratings after completing transactions, providing valuable insights for the community. Additionally, the platform encourages users to report unethical behavior or submit complaints, with all reports and feedback analyzed by the admin to maintain marketplace integrity.This marketplace does more than facilitate transactions—it builds trust. By integrating feedback loops, secure payments, and transparent processes, the system establishes itself as a reliable, user-centered environment. Its innovative features create a shopping platform that is not only efficient but also a safe space for users to engage in digital commerce confidently.

The project showcases how technology can create a harmonious blend of functionality, security, and accountability, setting a new standard for online peer-to-peer marketplace

CHAPTER 2

LITERATURE SURVEY

TITLE : Online Business and Marketplaces
AUTHORS : Yun Fong Lim, Kejia Hu
YEAR : 2024

This book serves as an indispensable guide for understanding and managing online businesses, offering readers comprehensive insights into the dynamics of digital platforms. Focusing on both retail and service marketplaces such as Amazon, Alibaba, Airbnb, and Didi, it delves into the foundational business models that power these global giants. Readers are introduced to critical concepts such as demand forecasting, price optimization, and the strategic integration of artificial intelligence (AI) and data analytics to enhance business performance. The book also addresses operational challenges, providing detailed discussions on logistics technologies and order fulfillment strategies that ensure efficiency and customer satisfaction. A particular emphasis is placed on overcoming hurdles in last-mile delivery, a vital yet challenging aspect of online retail that significantly impacts customer experience. In addition to exploring technical and strategic aspects, the book offers practical advice on scaling digital platforms and managing their complexities. It equips readers with tools to navigate the competitive landscape of online businesses, ensuring they can make informed decisions to adapt and thrive in a rapidly evolving market. By blending theory with actionable strategies, this book is a valuable resource for entrepreneurs, business leaders, and anyone keen to excel in the online business domain. Its rich insights and practical guidance make it an essential read for those looking to build, scale, or optimize digital platforms in today's interconnected world.

TITLE : Analysis of Online Marketplace for Local Vendors

AUTHORS : Thilak Raja P A, Mohammed M Iqbal, Dr. Umakanth S

YEAR : 2023

This paper delves into the evolving relationship between online marketplaces and local vendors, with a focus on small businesses and street vendors in Bangalore. It examines how digital platforms are reshaping the dynamics of local commerce, investigating key factors such as product quality, competitive pricing, customer service, and increased exposure that contribute to vendor success in the digital marketplace.

A notable innovation introduced in the research is a bidding system for product orders, designed to improve vendor performance while attracting a broader customer base. By enabling vendors to compete on price and service offerings, the bidding mechanism fosters healthy competition and ensures fair opportunities for small businesses to thrive. The study also explores the rapid growth of online grocery shopping and the increasing adoption of digital tools among consumers and vendors alike. It highlights how local vendors can utilize technology to extend their market reach, improve customer engagement, and sustain their businesses in an increasingly digital economy.However, the paper does not shy away from addressing the challenges faced by street vendors in urban environments, such as limited resources, lack of digital literacy, and competition with larger e-commerce platforms. By proposing the integration of bidding systems into online marketplaces, the study suggests a practical solution to bridge the gap between traditional vendors and the modern digital landscape. Through in-depth analysis and actionable recommendations, this paper provides valuable insights into the transformation of local commerce in Bangalore, emphasizing the importance of innovation and adaptability for local vendors navigating the digital era

TITLE : Seller Experience Assessment in Online Marketplace

AUTHORS : Amresh Kumar, Pallab Sikdar, Raiswa Saha

YEAR : 2021

This study explores the experiences of vendors operating on online marketplaces, with the goal of developing and validating a robust scale to measure seller satisfaction. As e-commerce continues its rapid growth, more businesses are leveraging digital platforms to expand their market presence. However, understanding and improving the seller experience remains a critical factor in ensuring sustainable marketplace success. The research identifies key factors influencing seller satisfaction through a multifaceted approach. It begins with an extensive review of existing literature, followed by in-depth interviews with vendors registered on major e-commerce platforms. Insights from marketing experts further refine the understanding of seller interactions, challenges, and expectations. The result of this investigation is a structured, validated scale that provides a comprehensive framework for assessing vendor satisfaction. This tool offers e-commerce platforms valuable insights into the strengths and weaknesses of their vendor engagement strategies. By addressing areas such as platform usability, support services, pricing structures, and overall vendor performance, the scale helps marketplaces identify opportunities for improvement. The study emphasizes the importance of fostering a positive vendor experience to maintain a healthy marketplace ecosystem. Satisfied vendors are more likely to remain active, offer high-quality products, and attract a loyal customer base, contributing to the overall growth of the platform. Furthermore, the findings encourage marketplace operators to adopt a vendor-first approach, focusing on continuous innovation and support mechanisms to strengthen vendor relationships. By bridging the gap between vendor expectations and platform capabilities, this research not only advances academic knowledge but also serves as a strategic guide for marketplace operators aiming to thrive in the competitive e-commerce landscape.

TITLE : Designing Online Marketplaces: Trust and Reputation

AUTHOR : Michael Luca

YEAR : 2017

This paper explores the pivotal role of trust and reputation mechanisms in shaping the structure and operation of online marketplaces. Platforms like eBay, Uber, and Airbnb rely heavily on review systems and information transparency to cultivate trust between buyers and sellers, which is essential for fostering successful transactions in the digital space. Reputation mechanisms, primarily through user reviews, serve as a cornerstone for these platforms by enabling participants to evaluate the trustworthiness of potential transaction partners. These mechanisms not only help users make informed decisions but also encourage high-quality behavior by making user actions visible and accountable. Despite their effectiveness, the paper highlights several challenges that threaten the integrity of these systems. These include **strategic review manipulation**, where users intentionally skew ratings for personal or competitive advantages, **selection bias**, as only certain users may choose to leave reviews, and distortions caused by **self-promotional or fake content**. Such issues undermine the credibility of review systems, potentially eroding user trust in the marketplace. Luca addresses these challenges with innovative design strategies aimed at improving the reliability of trust mechanisms. Recommendations include enhancing review verification processes to identify and filter out fake content, refining reciprocal review systems to reduce bias, and implementing selective information sharing to ensure that only relevant and accurate data is made available to users. By tackling these complexities, the paper provides actionable insights into how online marketplaces can maintain and strengthen their reputation systems. A robust trust framework is not only essential for safe and trustworthy transactions but also critical for the long-term growth and sustainability of digital marketplaces in an increasingly competitive landscape. Through these improvements, platforms can continue to build user confidence and enhance their overall functionality in the digital economy.

TITLE : The Social Infrastructure of Online Marketplaces

AUTHORS : Patrik Aspers, Asaf Darr

YEAR : 2022

This study investigates the "social infrastructure" underpinning online marketplaces, focusing on the intricate relationship between work and trade. Using Etsy as a key example, Aspers and Darr argue that these platforms operate as unique hybrid spaces where entrepreneurial labor and commerce intersect. This duality is orchestrated through both formal and informal systems that are constantly shaped and reshaped by the interactions of sellers and buyers. Formal systems, such as platform policies, payment mechanisms, and listing algorithms, provide the structural framework that governs marketplace activities. These rules and tools are designed by the platform to ensure smooth operations, secure transactions, and scalability. In contrast, **informal systems** emerge organically, driven by user behavior, community norms, and interpersonal dynamics. These informal systems include the social relationships sellers build with buyers, creative branding efforts, and the establishment of trust through personal engagement. The research highlights how these two systems coexist, often in tension but also in synergy, to create a dynamic and evolving marketplace environment. Sellers, for instance, navigate platform-imposed constraints while leveraging informal strategies to stand out in a competitive space. Buyers, on the other hand, rely on a mix of formal cues, such as ratings and reviews, and informal elements like personal interactions, to make purchasing decisions. Etsy's unique focus on handmade and vintage goods amplifies the importance of these informal practices, as sellers invest creative labor to differentiate their products while fostering personal connections with their customers. This interplay demonstrates how online marketplaces are not just transactional spaces but also socio-economic ecosystems where users actively participate in shaping their experiences. Ultimately, this research deepens our understanding of the evolving role of online marketplaces as spaces of economic activity and social interaction, offering practical recommendations for designing platforms that balance structure and adaptability to meet the complex needs of their user.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

There are several existing systems and technologies enabling online classified marketplaces, each focusing on specific functionalities such as product listing, payment processing, user communication, and search features.

E-COMMERCE PLATFORMS:

- eBay: It provides a robust platform for buying and selling items with secure payment integration and a feedback system.

PAYMENT SYSTEMS:

- PayPal**:** It facilitates secure online payments for marketplace transactions, providing buyer and seller protections.

REVIEW SYSTEMS:

- Trust Pilot: It aggregates user reviews, enhancing credibility and trust for both buyers and sellers.

PRODUCT LISTING AND INVENTORY MANAGEMENT SYSTEMS:

- Shopify: It offers a seamless platform for product listing, inventory management, and payment processing for sellers.

COMMUNICATION SYSTEMS:

- Email Notifications: It alert buyers and sellers about new messages, status updates, or confirmations related to their transactions.

3.2 PROPOSED SYSTEM

The proposed system for an online classified marketplace is designed to prioritize user trust, security, and engagement. A suite of features is integrated to ensure safe and seamless transactions while fostering a strong sense of community among users. Key among these enhancements is **email verification**, which validates buyers and sellers, significantly reducing fraudulent activities. To further secure transactions, an **escrow mechanism** will act as a transaction guarantee system. This ensures that funds are held securely until both parties fulfill their obligations, offering refunds when necessary to protect users. Location-based filters, powered by HTML5 geolocation, will allow users to discover products and services in their vicinity. This feature enhances user convenience by displaying relevant listings based on proximity. Simultaneously, event listeners will track product views, providing sellers with valuable insights into buyer interest and enabling a more personalized user experience. To address security comprehensively, a customer shield will be implemented. This protection feature includes scam prevention, reporting mechanisms, and tools for resolving fraudulent transactions. Alongside this, 24/7 customer support ensures constant assistance for users, reinforcing trust in the platform. A reservation system will also be introduced to prevent overlapping reservations by multiple users, ensuring a fair process for product acquisition. For improved community engagement, **forums** and Q&A sections will offer spaces for discussion, advice, and knowledge sharing among users. Communication will be further enhanced through **real-time** chat integration, utilizing services like Firebase or Intercom. This feature enables immediate, direct interaction between buyers and sellers, streamlining negotiation and improving overall user satisfaction. By combining these features, the proposed system creates a secure, user-centric platform where trust, functionality, and community engagement are at the forefront. These innovations aim to establish the marketplace as a reliable and efficient environment for conducting digital transactions.

3.3 BLOCK DIAGRAM OF PROPOSED SYSTEM

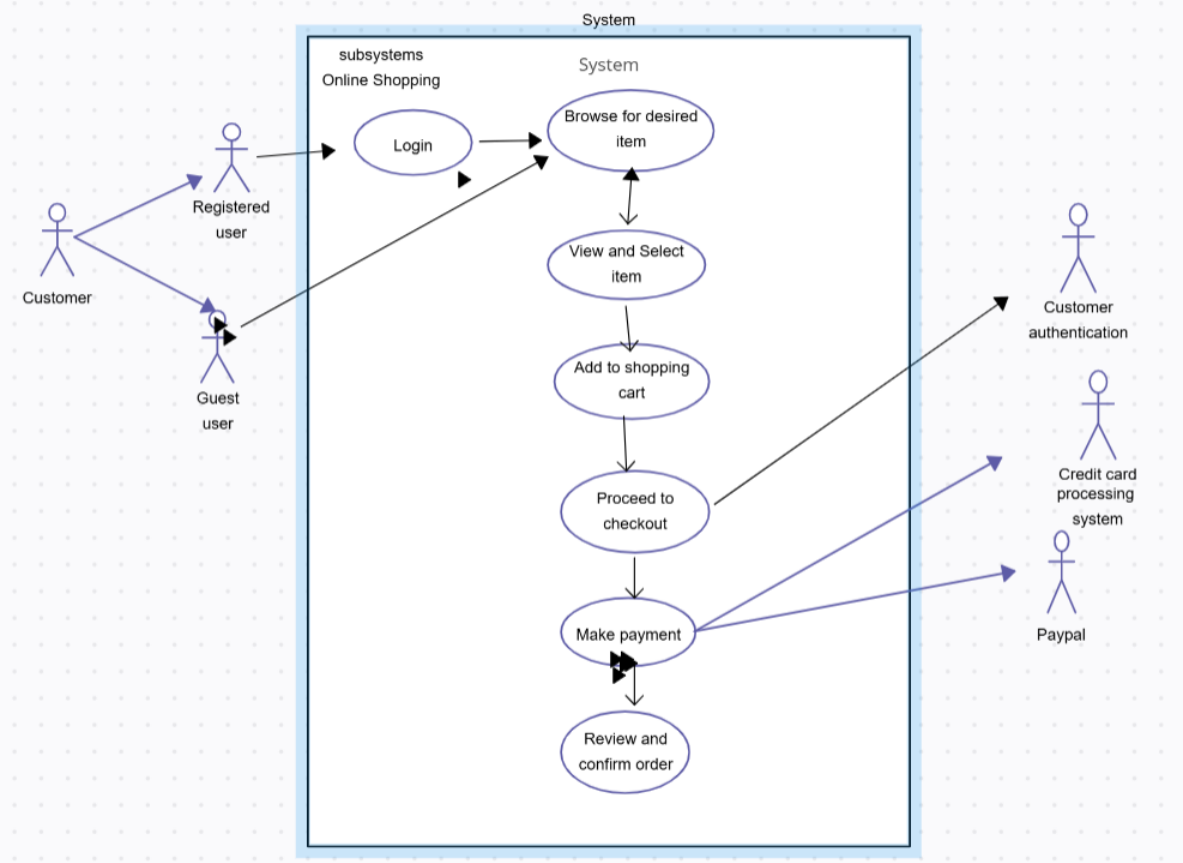

Figure 3.1: Usecase Diagram

3.4 FLOWCHART

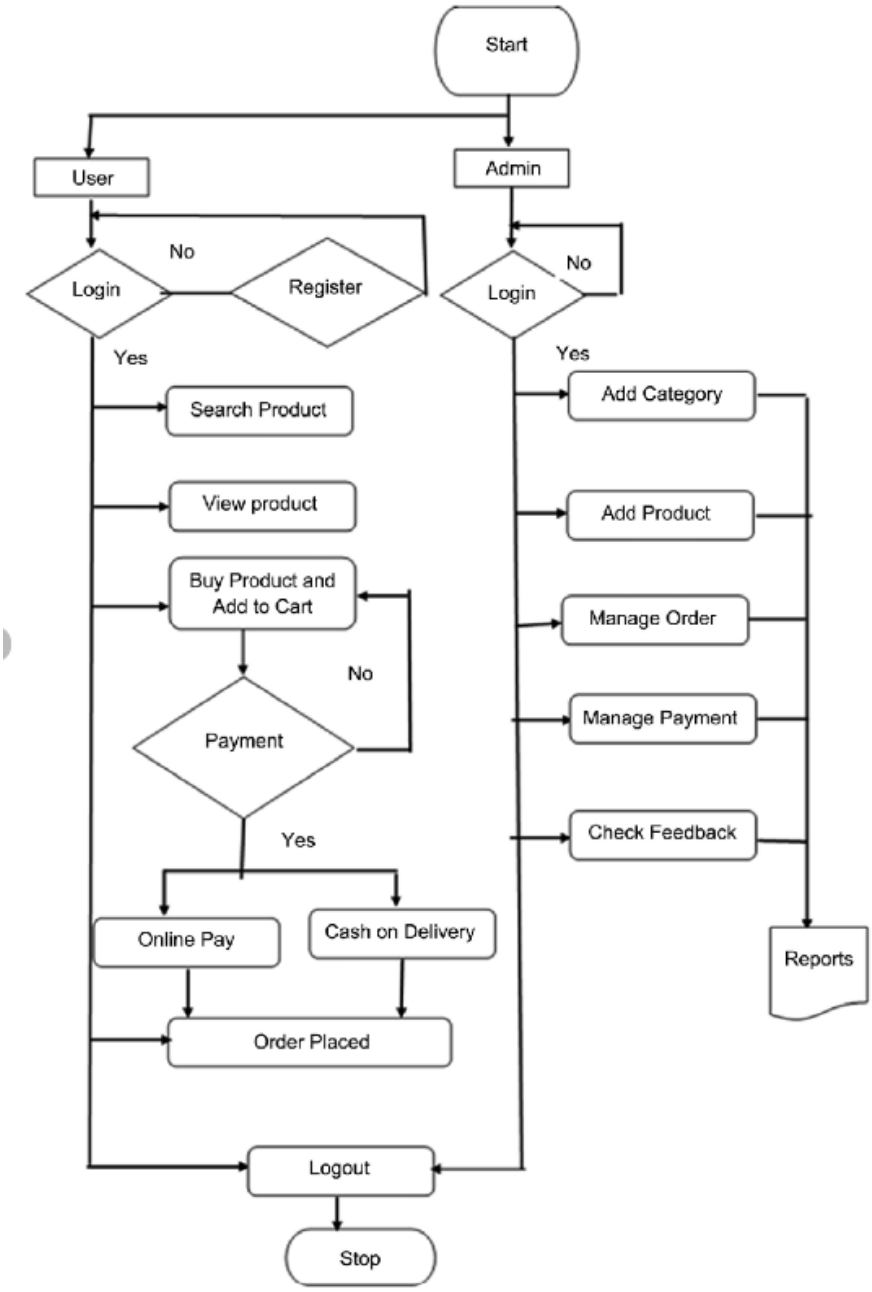

Figure 3.2: Flow Chart

3.5 PROCESS CYCLE

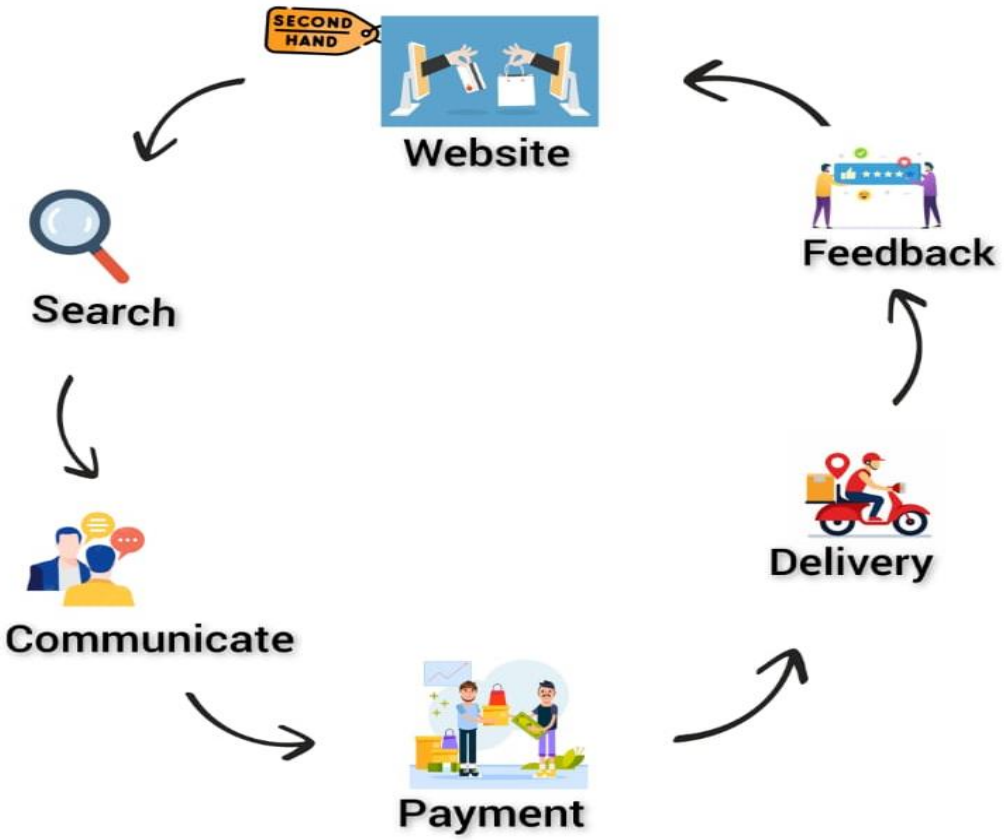

Figure 3.3:Process Cycle

3.6 ACTIVITY DIAGRAM

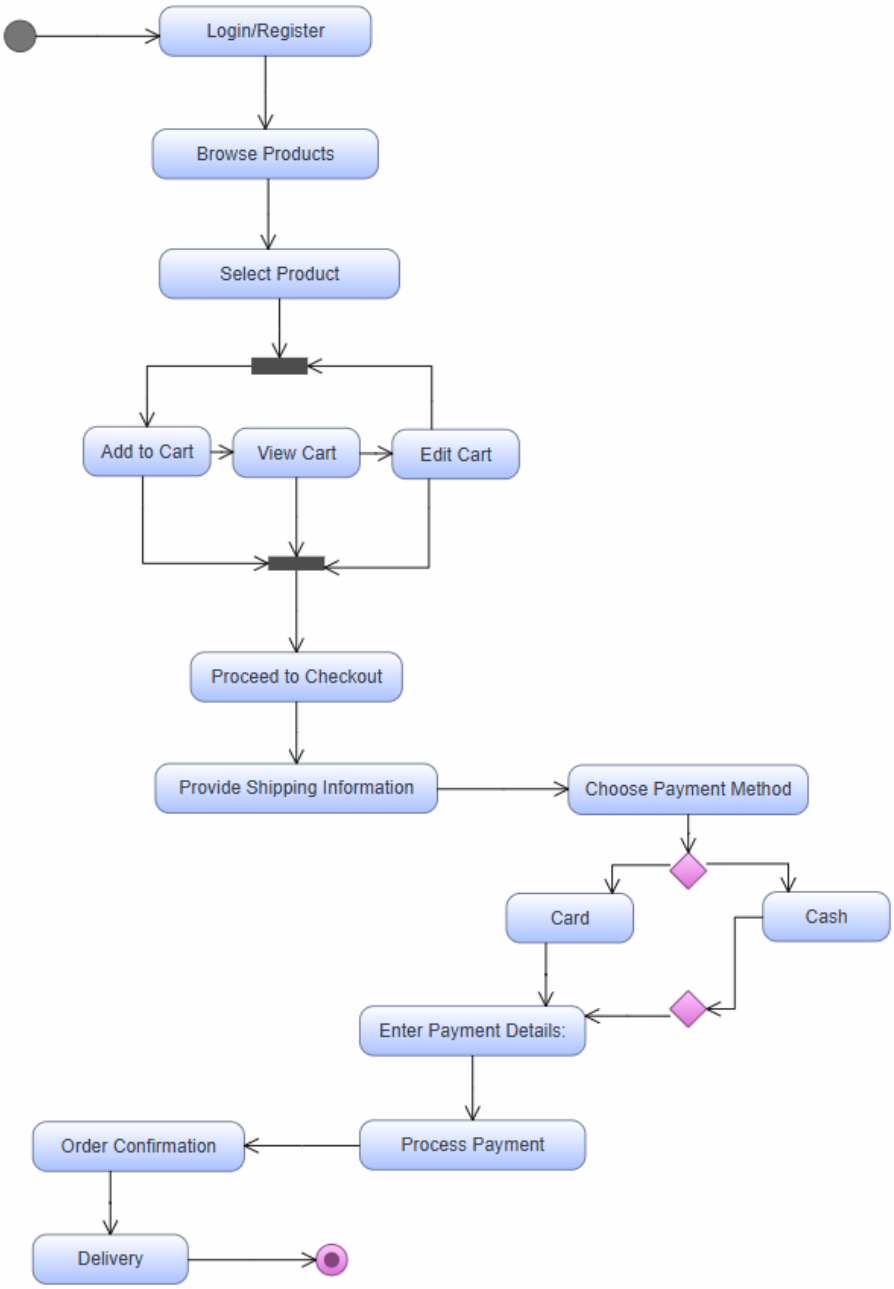

Figure 3.4: Action Sequence of process

3.7 CLASS DIAGRAM

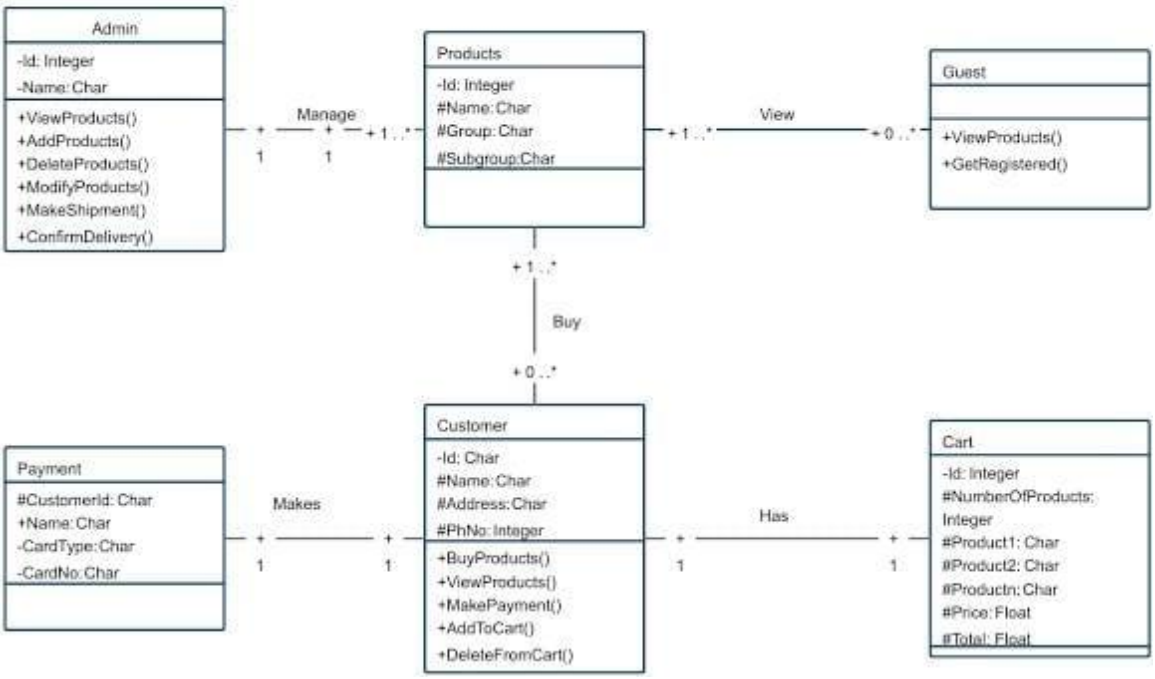

Figure 3.5: Class Diagram

CHAPTER 4

MODULES

4.1 MODULE DESCRIPTION

1. Account Management
2. Product Uploading & Searching
3. Buyer & Seller Interaction
4. Safe Transaction
5. Review & Complaints

4.1.1 ACCOUNT MANAGEMENT

The Account Management module is a crucial component of the online marketplace, responsible for managing user identities and ensuring secure access to the platform. It handles the entire user registration process, where users can sign up with their email addresses or other authentication methods, such as social media accounts. Once registered, the module supports authentication to verify the user's identity during login, ensuring that only authorized users can access their accounts. The password management functionality enables users to securely reset or change their passwords as needed. Additionally, the module stores and maintains user-specific data, such as personal preferences, transaction history, and profile settings, which help personalize the user experience and facilitate smoother transactions. By managing these aspects efficiently, the Account Management module ensures both security and ease of use for the platform's users.

1. Registration
2. Two - Step Verification
3. Login
4. Update Profile

4.1.1.1 REGISTRATION

The registration process in this system is designed to allow users to sign up with their email, user ID, phone number, and password. Upon clicking the "Register" button, the system first validates the input fields. Then, Firebase's authentication service is used to create a new user account based on the provided email and password. If the registration is successful, the user's email is verified by sending a confirmation email. Additionally, the system stores the user's user ID, email, and phone number in Firebase Firestore for future use. If any errors occur during registration, such as an already registered email or phone number, the system shows an appropriate error message via a popup. After successful registration and email verification, the user is redirected to the login page. This ensures a secure and streamlined registration process, with verification and error handling in place.

4.1.1.2 TWO -STEP VERIFICATION

The two-step verification process enhances account security by adding an extra layer of protection during user registration and login. After a user successfully registers, the system sends a verification link to the provided email address using Firebase's authentication service. This link serves as a confirmation that the email address belongs to the user, ensuring that the account is being registered by a legitimate person. Upon clicking the link, the user's email address is marked as verified, allowing them to fully access their account.This two-step verification process is a crucial security measure to prevent unauthorized access and to ensure that users can only interact with their account if they have access to the email address they provided. The verification process also helps in reducing fraudulent registrations and improving the overall integrity of the platform. If the user does not verify their email within a specific time frame, they may be unable to access some features of the platform until they complete the verification step, reinforcing the importance of this extra security check.

4.1.1.3 LOGIN

The login system in the code is designed to authenticate users with email or phone number and password. Upon entering their credentials, users can click the Login button to initiate the authentication process using Firebase's signInWithEmailAndPassword method. If the credentials are correct, the user is logged in successfully, and their user information (such as user ID and email) is stored in the browser's local storage for session management. The user is then redirected to the homepage (homepage.html), where they can access the platform's features. In case of a failed login attempt, an error message is displayed in a popup window to inform the user about the issue (e.g., incorrect email/password), ensuring transparency and a user-friendly experience.Additionally, the forgot password feature allows users to reset their password in case they have forgotten it. By clicking the "Forgot Password?" button, the system prompts the user to enter their email address, and a password reset link is sent via Firebase's sendPasswordResetEmail method. If the reset request is successful, the user is notified with a prompt to check their inbox for further instructions. If an error occurs (e.g., invalid email or network issues), an error message is shown in the same popup for immediate user attention. This feature improves the security and user experience by allowing users to regain access to their accounts without requiring direct support intervention.

4.1.1.4 UPDATE PROFILE

The User Profile section allows users to manage and update their personal information within the application. Upon successful authentication, the user’s profile page displays their unique user ID and email, and provides an option to upload or update their profile picture. The profile photo is uploaded directly to Firebase Storage and linked to the user’s Firestore document, ensuring the image is securely stored and easily accessible. Additionally, users can write or modify a personal description in the "about" section, which is stored in Firestore as part of the user’s profile data. This section ensures that users can personalize their profile, and any changes they make (such as

uploading a new picture or editing the "about" text) are immediately reflected in their account. If a user is not logged in, the system displays a prompt encouraging them to log in or register, ensuring that only authenticated users can access and update their profile information. The profile page also includes options to skip the update process and proceed to the homepage or return to the login page if they are not logged in. This setup fosters a more personalized and user-friendly experience, enhancing the overall functionality of the platform.

4.2 PRODUCT UPLOADING & SEARCHING

The Product Uploading feature in the platform allows users to easily add items for sale by filling out a simple form. Users can provide essential details such as product name, description, category, price, and product images. This ensures that all necessary information is captured and displayed in a structured manner, making the products easy to browse. Sellers can also select relevant categories and tags for their products, helping potential buyers find what they're looking for. Once uploaded, the product is immediately available for browsing, allowing sellers to connect with buyers quickly. The platform also allows sellers to update their product listings if needed, ensuring information remains accurate. The Product Searching functionality allows users to search for products efficiently based on various criteria. The search bar supports multiple filters such as product name, price range, or keywords. Users can also use tags to narrow down their search results. The results are displayed in a clear and organized manner, making it easier for buyers to compare and make informed decisions. This search functionality ensures that the platform is user-friendly, giving buyers a streamlined way to browse and purchase products while helping sellers gain exposure..

1. Product Listing
2. Product Searching
3. Product Liking

4.2.1 PRODUCT LISTING

The Product Listing feature enables sellers to upload and showcase their products on the platform. This process begins by filling out a comprehensive form, where sellers provide key details about the product, such as uploading an image, selecting the category it belongs to (e.g., audio, furniture, health & nutrition, etc.), and entering a descriptive title and description. Sellers must also set a price for their product and choose the listing type—whether it's a standard listing, a customer-protected listing, or an auction-style listing. This ensures that the item is categorized correctly, making it easier for potential buyers to find and explore the product. Additionally, a photo of the product is uploaded to visually attract buyers, creating a more engaging shopping experience.Once all the required fields are filled and the form is submitted, the product details, including the image, category, price, and description, are saved to the platform's database, making the listing visible to all users. The platform supports different listing types, such as Customer Protection, which ensures a safe transaction for both the buyer and seller, or a Normal Listing for standard sales. The auction-style listing allows users to bid on products, creating a competitive environment for high-demand items. The system also ensures that only authenticated users can post listings, safeguarding the platform from fraudulent activities. After submitting the listing, sellers are alerted with a success message, and the product is immediately available for browsing by buyers. The entire process is designed to be seamless, allowing sellers to reach a broad audience while ensuring the product details are displayed clearly and effectively.

4.2.2 PRODUCT SEARCHING

The search functionality in the application enhances the user experience by enabling users to easily find relevant products based on specific keywords. When a user performs a search, the query term entered in the URL's query parameter is retrieved and

converted to lowercase for case-insensitive comparison. The search checks multiple fields: the product's name, its type (e.g., auction, customer protection), and the user's email associated with the product listing. If any of these fields contain the search term, the product is displayed in the search results.The system also handles product availability by displaying distinct icons for products marked as "sold" or "reserved," allowing users to quickly distinguish between available and unavailable items. When displaying the search results, the application dynamically generates a product card for each matching product, showing key details like the product's name, description, price, category, and associated email. The product image and like button are also included, with real-time updates to the like count when users interact with it.If no products match the search term, a "No results found" message is shown, ensuring that the user knows the search didn’t return any results. This functionality offers a seamless way for users to browse products and find items based on specific keywords, making it easier for buyers .

4.2.3 PRODUCT LIKING

The product liking feature in the code allows users to express interest in a product by clicking a "like" button. When a user clicks the like button, the handleLike function checks if the product is already in the user's list of liked products stored in their Firestore document. If the product is already liked, the function removes it from the user's list and decreases the product's like count. If the product is not liked, it adds the product to the list and increases the like count. The updates are made to both the user's document (in the liked array) and the product's document (by modifying the likes field). This system allows for real-time updates of likes, ensuring that the product's like count is accurately reflected for all users, and provides a way for users to keep track of products they are interested in

4.3 BUYER & SELLER INTERACTION

In the buyer and seller interaction feature, the application facilitates communication between users via private, group, and protected chat systems. The chats, groupChats, and protectedChats collections in Firebase Firestore store the details of individual, group, and protected conversations. Each chat includes users' IDs and the associated product details, enabling users to discuss products, negotiate prices, or finalize deals. For each type of chat (individual, group, protected), the code fetches chat data using queries like where('users', 'array-contains', userId) to retrieve conversations involving the authenticated user. Once the data is fetched, the application displays the chats with details such as the last message exchanged and product information, including a product image, name, and description. The code also includes functionality to link to specific chat pages (chat.html, group-chat.html, protected-chat.html), where users can continue discussions related to a particular product. This interaction enables buyers and sellers to engage with each other in a secure and organized manner.

1. Normal Chat
2. Auction Chat
3. Customer – Shield Chat

4.3.1 NORMAL CHAT

The normal chat feature in this application allows buyers and sellers to communicate within a product listing. When a user accesses a chat, the system retrieves the chat messages for that specific chat ID from Firebase Firestore. Using the fetchChatMessages function, messages are displayed in real-time through onSnapshot, which listens for new updates in the chat. Each message is shown along with the sender's profile photo and email, and the messages are ordered chronologically. The

sendMessage function allows users to send messages to the chat, which are stored in the Firestore collection corresponding to the chat ID. This creates an ongoing conversation that both buyers and sellers can use to negotiate, ask questions, or finalize deals. The interface dynamically updates the chat messages and scrolls to the latest message, providing a seamless communication experience. Additionally, the application checks for user roles (buyer or seller) to tailor the interface and options accordingly, such as showing relevant buttons for making offers or accepting them.

4.3.2 AUCTION CHAT

The auction chat functionality in the provided code enables real-time interaction between buyers and sellers in an online marketplace. When a buyer makes an offer on a product, they can send a message with the offer amount in the chat. This offer is then recorded in a Firestore collection called auctions, where the details of the offer are stored, including the product ID, user ID, offer amount, and timestamp. The seller, upon receiving the offer, can either accept or reject it. If the seller accepts the offer, the system updates the product details with the buyer's information and sets the offer as accepted in the chat. Additionally, a review document is created to allow both the buyer and seller to leave feedback for each other. The chat interface also provides options for both the buyer and seller to reserve or unreserve the product, make offers, and accept offers. The dynamic interaction is supported by event listeners that handle actions like sending messages, updating product availability, and displaying review containers after the offer is accepted.

4.3.3 CUSTOMER – SHIELD CHAT

The customer-shield chat functionality facilitates communication between buyers, sellers, and admins in a secure, role-based environment. The system differentiates user roles—buyer, seller, and admin—by monitoring user IDs and adjusting UI elements accordingly. Buyers can make offers, add products to their cart, and confirm the receipt of products, while sellers can reserve/unreserve products, accept offers, and update the shipment status. Admins are able to intervene in the chat if requested by either the buyer or the seller, enabling them to oversee the transaction. Each action, such as confirming a deal or shipping a product, triggers updates to the database, ensuring the transaction progresses smoothly. Additionally, users can submit reviews once a transaction is completed. The chat dynamically adjusts to reflect the transaction status, ensuring that all parties involved are informed and engaged at every step of the process.

4.4 SAFE TRANSACTION

The Safe Transaction module of our platform ensures secure and reliable exchanges between buyers and sellers by acting as an intermediary throughout the transaction process. When a buyer adds items to their cart and confirms the purchase, the transaction status is updated to "pending," and the admin is notified to oversee the transaction. The admin’s role is crucial as they verify that the seller has dispatched the product before the status is changed to "completed." Only after this confirmation is the payment processed. The payment method used is UPI (Unified Payments Interface), a fast and secure method that guarantees a seamless transaction between the buyer and the platform. However, the buyer’s payment is only fully processed once the admin confirms that the product has been shipped, ensuring that the buyer's money is protected until they receive the product. This process is designed to protect both the buyer, who can be assured that they will receive the product as promised, and the seller, who is guaranteed payment once the product is dispatched. The admin's oversight plays a key role in maintaining the integrity of each transaction, preventing fraud, and ensuring that

sellers are held accountable for timely shipments. To further protect users, the system includes a **Customer Shield** feature that offers scam protection, enabling users to report fraudulent activity and resolve any disputes quickly. The platform also provides 24/7 customer support, which ensures that users can access assistance whenever they need it. The **reservation system** is another key feature that prevents multiple buyers from reserving the same product at the same time, ensuring that each buyer has a fair chance of securing the product they wish to purchase. This system creates a seamless and transparent experience for both buyers and sellers, offering peace of mind by ensuring that both parties' interests are safeguarded throughout the transaction process. By acting as a trusted intermediary and using secure payment methods like UPI, the platform builds a robust framework for safe transactions that fosters trust between buyers and sellers and helps reduce risks such as fraud or disputes. The Safe Transaction module is designed to provide a secure, reliable, and user-friendly environment, encouraging users to engage in digital commerce with confidence, knowing their transactions are protected at every step.

4.5 REVIEWS & COMPLAINTS

The Review System in the project not only facilitates transparency and trust but also encourages a culture of honesty and constructive feedback. After a successful transaction, both buyers and sellers can rate each other on various aspects such as product quality, delivery time, communication, and overall satisfaction. This feedback is made visible to other users, allowing them to assess the reputation of potential transaction partners. The dual review mechanism is designed to foster mutual accountability and fairness, as both parties are equally encouraged to provide honest evaluations. This system ultimately builds a reputation-based framework that improves the overall user experience. The Complaint System further strengthens the platform by providing users with a formal process to report issues such as fraud, poor service, or unethical behavior. Once a complaint is submitted, the system notifies administrators, who can investigate and take appropriate action, such as issuing warnings, suspending accounts, or resolving disputes.

CHAPTER 5

SYSTEM SPECIFICATION

5.1 REQUIREMENTS

1. Firebase CLI
2. Firebase Hosting
3. Firestore and Realtime Database

5.2 SOFTWARE REQUIREMENTS

1. Processor – Intel i3 or Higher.
2. RAM – 4GB or Higher.
3. Storage – 150GB or Higher.

5.1.1 FIREBASE CLI

The Firebase Command Line Interface (CLI) is a powerful toolset that allows developers to interact with Firebase services directly from the terminal. It simplifies the process of configuring and deploying Firebase services, such as Firebase Hosting, Cloud Functions, Firestore, and Realtime Database. By using the CLI, developers can manage Firebase projects, test their code locally, deploy web apps, and handle various configurations, all from the command line. The Firebase CLI is essential for building and managing scalable applications on Firebase, as it enables seamless integration with Firebase's backend and deployment services. In this project, Firebase CLI plays a central role in deploying the application and managing Firebase services. First, it is used to initialize the project, linking it to Firebase Hosting and setting up configurations for

Firestore and Realtime Database. The CLI enables local testing, allowing the project to be previewed in a local environment before deployment. Finally, Firebase CLI handles the deployment process, pushing updates and new versions of the application to Firebase Hosting with a single command. By streamlining these operations, Firebase CLI enhances efficiency and supports continuous development and deployment throughout the project lifecycle.

5.1.2 FIREBASE HOSTING

Firebase Hosting is a secure, fast, and reliable web content hosting service provided by Google Firebase. It allows developers to deploy static web content, like HTML, CSS, and JavaScript, as well as dynamic content through server-side APIs and cloud functions. With Firebase Hosting, developers get free SSL certification, making all connections secure by default. The hosting service is optimized for speed, with content served from a global content delivery network (CDN) that ensures minimal latency by caching content across multiple locations worldwide. This hosting solution is well-suited for single-page applications, progressive web apps, and websites with real-time data needs due to its integration with Firebase's other backend services. In this project, Firebase Hosting serves as the deployment platform for the application, providing a reliable environment for hosting both static and dynamic content. After developing the application's HTML, CSS, and JavaScript components, Firebase Hosting is used to deploy the frontend to a secure, globally accessible URL. Firebase CLI facilitates this by managing and deploying updates to Firebase Hosting with just a few commands. Additionally, Firebase Hosting works in tandem with Firestore and Realtime Database to deliver real-time data updates, allowing users to interact seamlessly with the application's database. This integration enhances the user experience by providing quick, responsive access to hosted content and real-time data. Firebase Hosting also provides built-in versioning and rollback features, allowing developers to easily manage and revert to previous versions if necessary. It integrates smoothly with Firebase's powerful suite of tools, including Firebase Analytics, Firebase Authentication, and Firebase Cloud Functions, to create a seamless and scalable app

deployment process. The platform supports automatic deployment pipelines, ensuring that updates can be pushed live with minimal effort. With robust security features, including automatic HTTPS and easy configuration for custom domains, Firebase Hosting ensures secure, scalable, and efficient hosting for applications at every stage of development.

5.1.3 FIRESTORE AND REALTIME DATABASE

Firestore and Realtime Database are Firebase's two NoSQL database solutions for storing and syncing data. Firestore (Firebase Cloud Firestore) is a scalable, flexible database designed for structured data, with capabilities for complex querying and real-time data syncing across client applications. It is well-suited for structured data collections, with support for sub-collections, complex queries, and offline data access. Realtime Database, on the other hand, is Firebase's original database designed for simpler, hierarchical data. It offers ultra-fast data syncing and works especially well for applications requiring low-latency updates, like chat and collaboration features. Both databases offer real-time data synchronization and support offline persistence, making them ideal for responsive applications with continuous data updates. This project utilizes both Firestore and Realtime Database to manage and sync user data effectively. Firestore is used to handle structured collections, such as product listings, user profiles, and transaction records, allowing for powerful querying and scalability. By using Firestore, the application can easily retrieve and store data in collections, making it efficient to manage data like product information, reviews, and reports. RTDB, in contrast, is employed where low-latency data updates are essential, such as in real-time chat features. This enables users to engage in live conversations about products, with updates appearing instantaneously on both ends. Combining these two databases enhances the app's responsiveness, ensuring that both structured and real-time interactions are seamlessly managed. Additionally, Firestore supports complex data models with nested collections, offering flexibility for managing relationships between various data types. Realtime Database is ideal for low-latency syncing, crucial for features like live notifications and activity feeds. Both databases are highly scalable, allowing the app to handle increasing data loads as user activity grows.

CHAPTER 6

METHODOLOGY

6.1 TECHNOLOGY STACK

The technology stack for the project involves a combination of frontend and backend technologies to deliver a seamless user experience and robust functionality. For the frontend, HTML, CSS, and JavaScript are utilized to create the user interface and ensure responsiveness across different devices. HTML provides the structure for web pages, CSS handles the styling and layout, and JavaScript enables interactivity and dynamic content. On the backend, Firebase services are employed to handle authentication, data storage, real-time interactions, and hosting. Firebase Authentication ensures secure user login UID and management, while Firestore and Realtime Database are used to store user data, product details, messages, and transaction records. The Firebase CLI is used for managing the Firebase project and deploying the application, making it easy to push updates to the hosted environment. This stack ensures scalability, reliability, and a smooth experience for users on the platform.

6.2 ENVIRONMENT SETUP

The environment setup for the project involves configuring the necessary tools and services to ensure smooth development, testing, and deployment. The first step is setting up Firebase as the backend service, where the Firebase project is created through the Firebase Console. Firebase Authentication is configured to handle secure user login, and Firestore and Realtime Database are set up to manage data storage for user profiles, product listings, and chat interactions. Next, the frontend development environment is set up using a text editor like VS Code or Sublime Text, where the HTML, CSS, and JavaScript files are structured. Local development can be tested using a web browser to preview changes made to the frontend, with any dynamic content rendered via JavaScript interacting with Firebase. To manage the deployment and ensure that updates

are properly reflected in the live application, Firebase CLI is installed and configured, allowing developers to deploy the project directly to Firebase Hosting. This also helps with monitoring and managing Firebase services from the command line. Additionally, version control using Git is incorporated to track changes and collaborate on code, ensuring that the project is maintained efficiently and can be rolled back if needed. The setup process also includes configuring Node.js for JavaScript-related server-side tasks, ensuring compatibility with Firebase functions and dependencies. Overall, this environment setup allows for a streamlined development workflow, from local development to deployment on Firebase Hosting, ensuring smooth functionality and scalability of the project.

6.3 DATA AUTHENTICATION AND SECURITY

Data authentication and security form a crucial part of the project's design, ensuring that user data and transactions are protected. Firebase Authentication is implemented to manage secure sign-in options, supporting email-password logins. This setup restricts access to sensitive areas of the platform, allowing only verified users to perform actions like viewing product details, making transactions, and engaging in chats. Firebase Authentication also provides automatic handling of authentication tokens, simplifying secure access to Firebase services without manually managing user sessions. Data security extends to Firebase's database setup, where Firestore and Realtime Database implement strict read and write rules based on user roles and permissions. This structure protects sensitive data, ensuring that each user can only access their own data while preventing unauthorized access. For instance, buyer-seller communication is protected, and transactions are encrypted for privacy, minimizing risks during data transmission. Additionally, Firebase Hosting includes SSL certificates, ensuring encrypted data transfers between the user’s browser and the server. To further secure user data, the application employs Firebase’s built-in analytics and monitoring tools, enabling real-time detection and alerting of potential security threats or anomalies, helping maintain a safe, trustworthy environment for user interactions and transactions on the platform.

6.4 DATABASE STRUCTURE

The database structure is carefully organized to optimize data storage, retrieval, and management, ensuring a seamless experience for users. Firebase Firestore and Realtime Database are both employed, each serving distinct functions to handle the diverse data requirements of the platform. Firestore is primarily used for storing structured data related to user profiles, product listings, transactions, and reviews, leveraging its document-based model for quick access and scalability. Each collection within Firestore—such as "Users," "Products," "Transactions," and "Reviews"—contains documents where fields are tailored to store relevant details like user information, product descriptions, prices, transaction statuses, and user feedback. The Realtime Database is integrated to support live functionalities such as chat and notifications. Its real-time data syncing allows chat messages, notifications, and updates to be instantly pushed to users, facilitating smooth, real-time interactions between buyers and sellers. The structure includes collections like "Messages" and "Notifications," designed for efficient read and write operations to minimize latency and improve responsiveness. Access control and security rules are established within both databases to enforce user permissions, ensuring that each user can only access authorized data, such as their personal chat history or transaction records. This structured approach in Firestore and Realtime Database, along with Firebase's built-in security protocols, provides a scalable, efficient, and secure data management system tailored to meet the project's requirements for real-time engagement, user data security, and transactional integrity.

6.5 REAL - TIME CHAT IMPLEMENTATION

The real-time chat implementation is a key feature that enables users to communicate seamlessly within the platform, fostering interactions between buyers and sellers for effective deal-making. Using Firebase Realtime Database, chat functionality is designed to provide instantaneous messaging, where any new message is automatically synchronized and updated across both users' devices in real time. This

live sync capability minimizes latency, allowing users to exchange messages without delay and enhancing the interactive experience. Each conversation is stored in a structured format within the Realtime Database, typically under unique nodes for each user pair, such as "User1-User2," to organize message threads. Each message entry includes metadata like the message content, timestamp, and sender ID to ensure chronological order and accurate message attribution. To improve security and privacy, Firebase rules restrict access so that only the involved users can view their conversation, preventing unauthorized access to personal messages. Additionally, the chat feature incorporates indicators for message read status and typing status, further enhancing user engagement and usability. Firebase's scalable infrastructure supports this by efficiently managing concurrent connections, even under high user activity. Through this setup, real-time chat becomes an effective, reliable, and secure means of communication on the platform, providing users with a streamlined interaction channel for inquiries and negotiations directly related to listings and transactions.

6.6 TRANSACTION AND PAYMENT WORKFLOW

The transaction and payment workflow is carefully designed to ensure secure, smooth, and transparent transactions between buyers and sellers on the platform. Once a buyer selects an item to purchase, the transaction process initiates, guiding the user through a step-by-step checkout process that includes item confirmation, payment method selection, and order finalization. Firebase Firestore handles transaction data, logging each order's details such as item information, buyer and seller IDs, transaction amount, and status updates (e.g., "Pending," "Processing," "Completed"). This structure maintains a clear audit trail and ensures every transaction is accurately recorded. Payment processing integrates secure methods, with UPI (Unified Payment Interface) serving as the primary option. The platform does not handle or store sensitive payment data directly; instead, it relies on Firebase's secure environment and third-party UPI integration to safeguard payment details. To minimize fraud, Firebase Authentication restricts transactions to verified users only, ensuring that buyers and sellers are authenticated before initiating any financial exchange. Once the payment is confirmed, the platform automatically notifies the seller to prepare the item for delivery. The

transaction status in Firestore is updated accordingly, and both buyer and seller can track the progress. For added security, admin oversight is in place to mediate disputes and confirm successful transactions, enhancing trust and protecting both parties. This workflow ensures that transactions are secure, traceable, and user-friendly, contributing to a positive experience on the platform.

6.7 DEPLOYMENT AND HOSTING

The deployment and hosting process ensures that the platform is accessible, stable, and secure for users. Firebase Hosting is utilized for its reliable, scalable infrastructure and ease of integration with other Firebase services. During the deployment phase, Firebase CLI is employed to set up and deploy the application, allowing for quick uploads of frontend and backend code to Firebase's servers. This streamlined setup provides a fast, efficient way to push updates, enabling continuous development and improvement of the platform. Firebase Hosting offers secure connections by default, using SSL certificates to encrypt all data transmitted between users and the server. This secure environment is critical for protecting sensitive information, particularly during transactions and user authentication processes. Hosting static assets, such as HTML, CSS, JS, and image files, Hosting static assets, such as HTML, CSS, JS, and image files, ensures that the platform loads quickly and efficiently for users. Firebase Hosting's global content delivery network (CDN) caches these assets in multiple locations worldwide, minimizing latency and improving load times. This is particularly beneficial for users in different geographic regions, as they can access content from the nearest server location. Additionally, Firebase Hosting automatically handles the scaling of resources to accommodate varying traffic loads, ensuring that the platform remains responsive even during high-traffic periods. Firebase's integration with Firestore and Realtime Database allows dynamic content to be served seamlessly, offering real-time updates to users. With Firebase Hosting's easy-to-use deployment tools, updates and bug fixes can be applied with minimal downtime, keeping the platform up-to-date and reliable. This combination of speed, scalability, and security makes Firebase Hosting an ideal choice for maintaining a stable online platform.

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENT

7.1 CONCLUSION

In conclusion, this project serves as a comprehensive online classified marketplace aimed at facilitating smooth, secure, and efficient interactions between buyers and sellers. By leveraging the power of Firebase services, the platform ensures a user-friendly experience, supported by a robust backend for authentication, real-time database management, and reliable hosting. The use of Firestore and Realtime Database provides an efficient solution for managing and syncing data, enabling secure transactions and protecting user data with multiple layers of security. Through real-time updates and data encryption, the system ensures that both buyers and sellers are confident in their interactions, with transaction processes securely facilitated. The real-time chat functionality further enhances communication between users, allowing for seamless negotiations and the establishment of trust, which is essential in an online marketplace. Each feature of the platform is meticulously tailored to meet the needs of users, ensuring a smooth and reliable experience while prioritizing data privacy and security. With structured access controls in place, sensitive user data is protected, and encrypted data transfers guarantee secure communication between all parties involved. The project's modular approach ensures that every component, from profile management to transaction workflows, operates seamlessly within the overall ecosystem, creating a unified and intuitive platform. Firebase Hosting enables fast, secure, and efficient access to the platform, ensuring that users can engage with the marketplace without concerns over stability or performance. As the platform grows and scales, Firebase's scalable infrastructure allows for effortless adaptation, ensuring that the system can handle increasing demand while maintaining the same level of performance. Moreover, Firebase's deployment tools streamline the process of pushing updates and making improvements, ensuring the platform stays up-to-date with minimal downtime. This project not only addresses the current market demands but also lays the

foundation for future enhancements, such as integrating AI-driven search recommendations or implementing advanced analytics to further improve the user experience. Ultimately, this platform offers a scalable, secure, and interactive marketplace designed to empower users and simplify online transactions, providing a solid basis for continuous growth and future success.

7.2 FUTURE ENHANCEMENT

AI-Powered Product Recommendations: Implementing machine learning algorithms for AI-powered product recommendations will enhance user experience by personalizing the marketplace. These algorithms can analyze a user's past searches, browsing history, and purchase behavior to suggest products that align with their preferences. The system would continuously learn from user interactions, improving recommendations over time and ensuring that users are always presented with relevant products. This could lead to higher engagement and increased sales as users are more likely to explore and purchase items that are tailored to their needs and tastes. Advanced Search Functionality and Smart Price Estimation Tool: By incorporating advanced search filters and sorting options, users will be able to find products faster and with greater accuracy. Filters like price range, product category, location, and ratings would allow users to narrow down their choices more effectively. Additionally, a smart price estimation tool could help sellers set competitive prices by analyzing market trends, demand, and competitor prices. This would encourage sellers to price their products appropriately, ensuring that items are neither too expensive nor too cheap, ultimately benefiting both buyers and sellers in the marketplace. Chatbot for Customer Support and Fraud Detection System: Introducing an AI-powered chatbot would streamline customer support by providing immediate assistance to users. It could handle common inquiries, guide users through transactions and offer. It could also assist with troubleshooting issues, ensuring that users receive timely responses. Additionally, the integration of a fraud detection system powered by machine learning would help identify and prevent fraudulent activities, ensuring the safety and trustworthiness of the platform.

APPENDIX – 1

SOURCE CODE

login.js

```
const firebaseConfig = {

    apiKey: "AIzaSyAm9VTkFRrLYC2jnT3xAy8kwj2cAjLjyI4",

    authDomain: "projj-7fa1d.firebaseapp.com",

    projectId: "projj-7fa1d",

    storageBucket: "projj-7fa1d.appspot.com",

    messagingSenderId: "731552517830",

    appId: "1:731552517830:web:10447552198f5af07c91c7",

    measurementId: "G-H5JBP27FBD"

};



const app = firebase.initializeApp(firebaseConfig);

const auth = firebase.auth();



const loginBtn = document.getElementById('loginBtn');

const forgotPassword = document.getElementById('forgotPassword');

const errorPopup = document.getElementById('errorPopup');
```

```
const overlay = document.getElementById('overlay');
const errorMessage = document.getElementById('errorMessage');
const closePopup = document.getElementById('closePopup');

loginBtn.addEventListener('click', () => {
    const emailOrPhone = document.getElementById('emailOrPhone').value;
    const password = document.getElementById('password').value;
    firebase.auth().signInWithEmailAndPassword(emailOrPhone, password)
        .then((userCredential) => {
            const user = userCredential.user;
            localStorage.setItem('userID', user.uid);
            localStorage.setItem('userEmail', user.email);
            window.location.href = "homepage.html";
        })
        .catch((error) => {
            errorMessage.innerText = error.message;
            errorPopup.style.display = 'block';
            overlay.style.display = 'block';
        });
```

```
});

forgotPassword.addEventListener('click', () => {
    const emailOrPhone = prompt('Please enter your email to reset your password:');
    if (emailOrPhone) {
        firebase.auth().sendPasswordResetEmail(emailOrPhone)
            .then(() => {
                alert('Password reset email sent. Please check your inbox.');
            })
            .catch((error) => {
                errorMessage.innerText = error.message;
                errorPopup.style.display = 'block';
                overlay.style.display = 'block';
            });
    }
});

closePopup.addEventListener('click', () => {
    errorPopup.style.display = 'none';
    overlay.style.display = 'none';
});
```

APPENDIX – 2

SCREENSHOTS

Sample Output

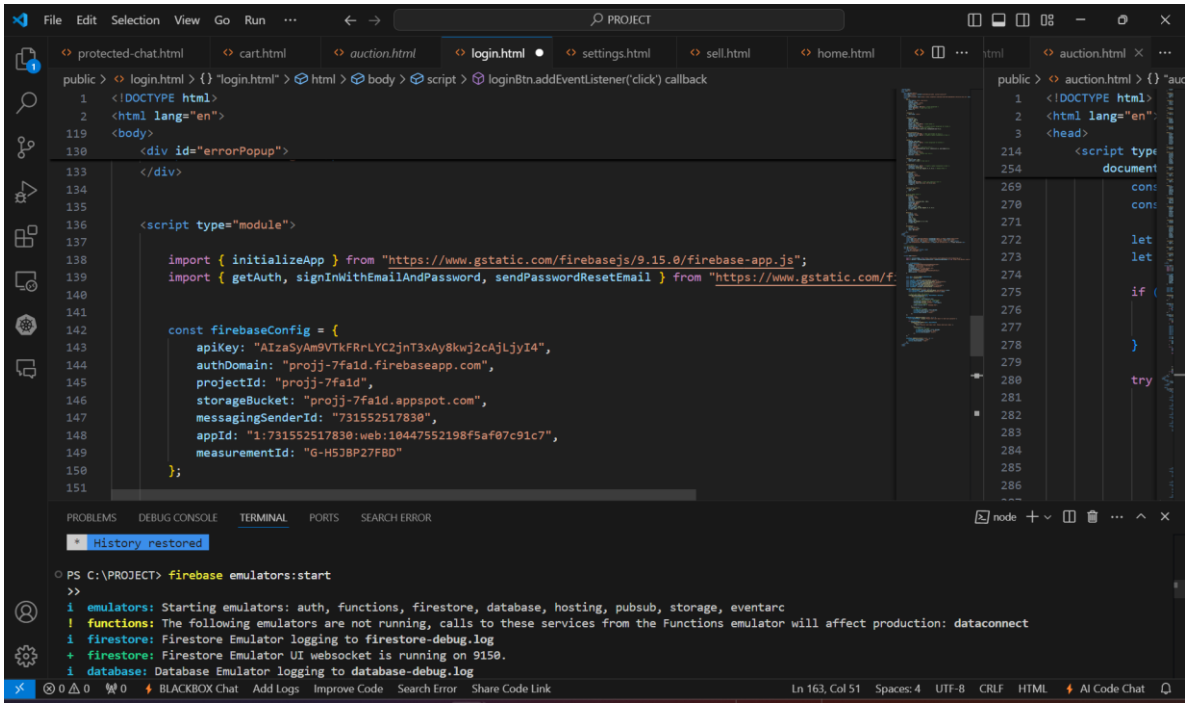

Figure 2.1: Execution of code

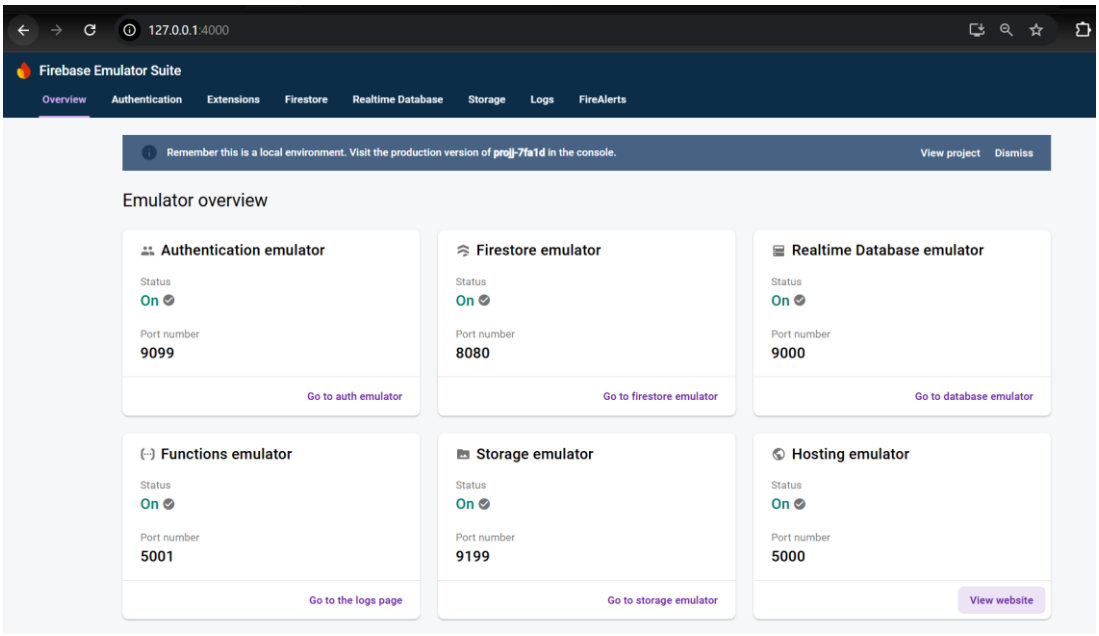

Figure 2.2: Firebase Emulator Suite

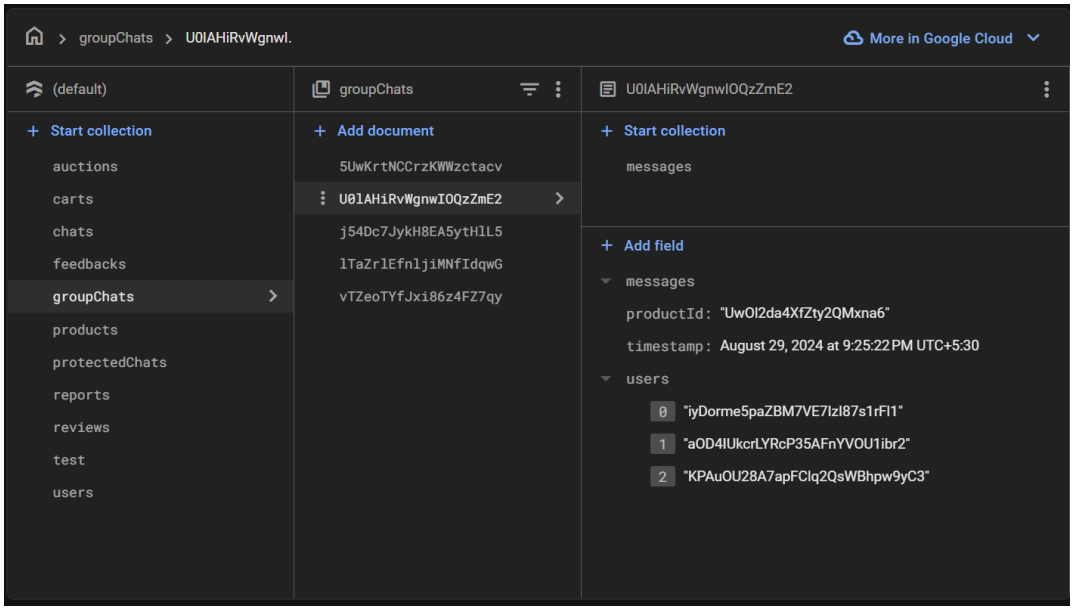

Figure 2.3: Cloud Firestore Database

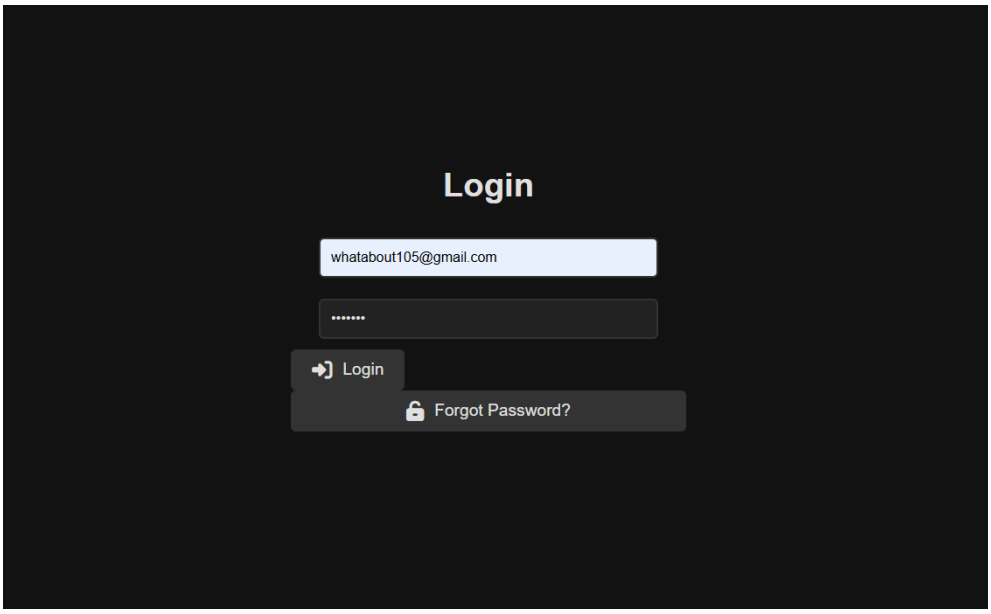

Figure 2.4: Login

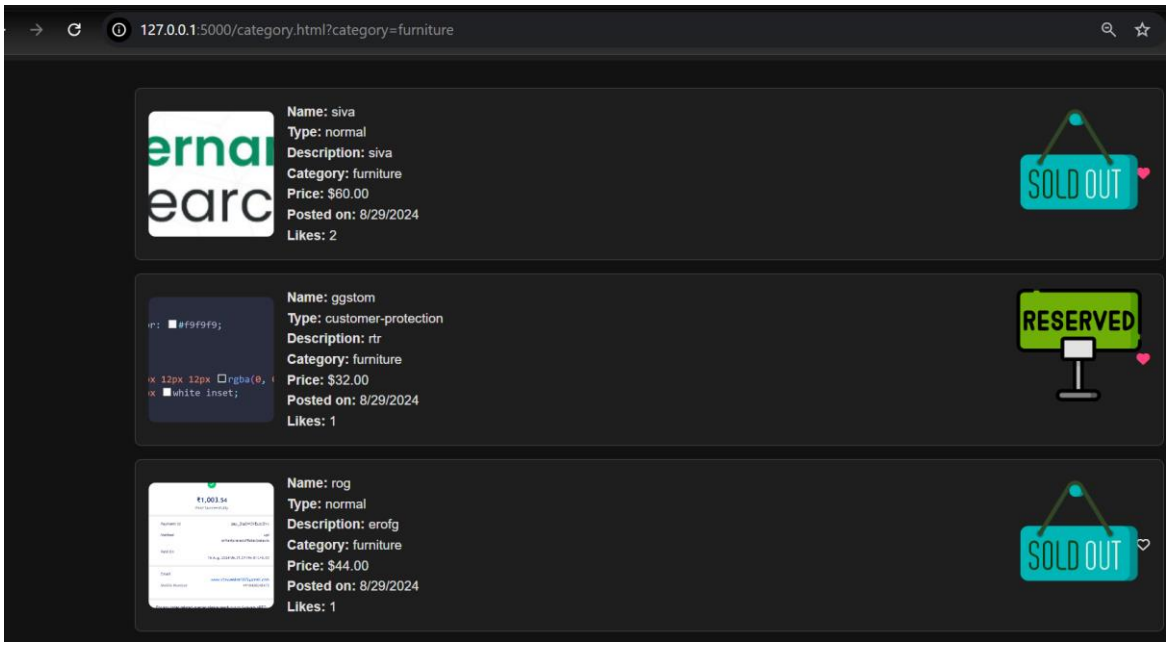

Figure 2.5: Product Searching

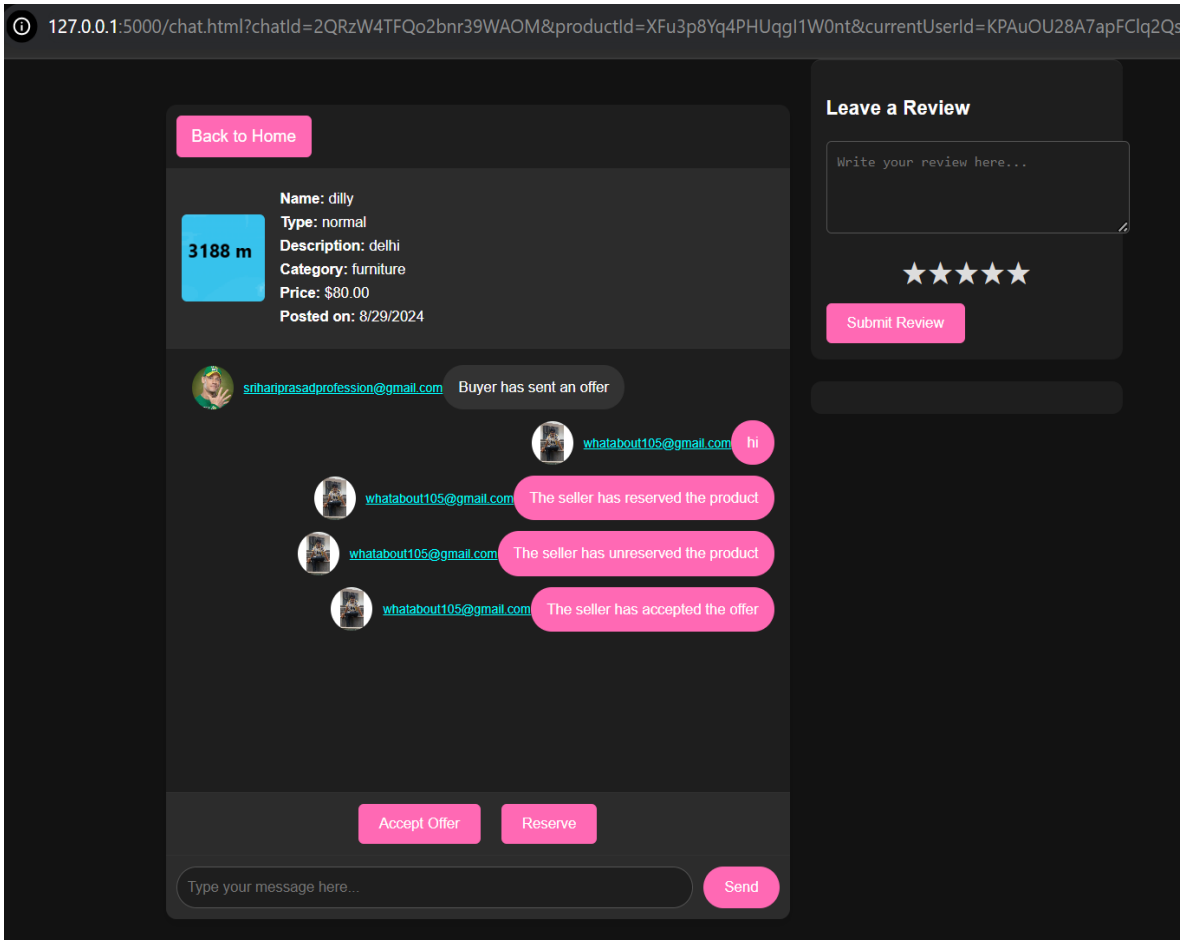

Figure 2.6: Customer Chat

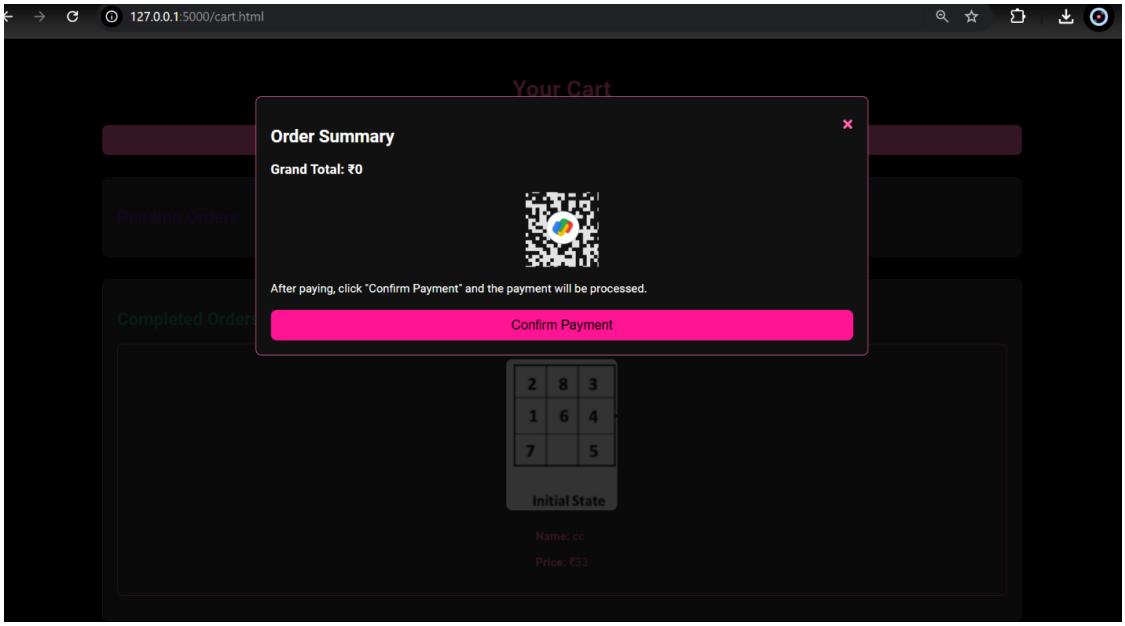

Figure 2.7: Payment

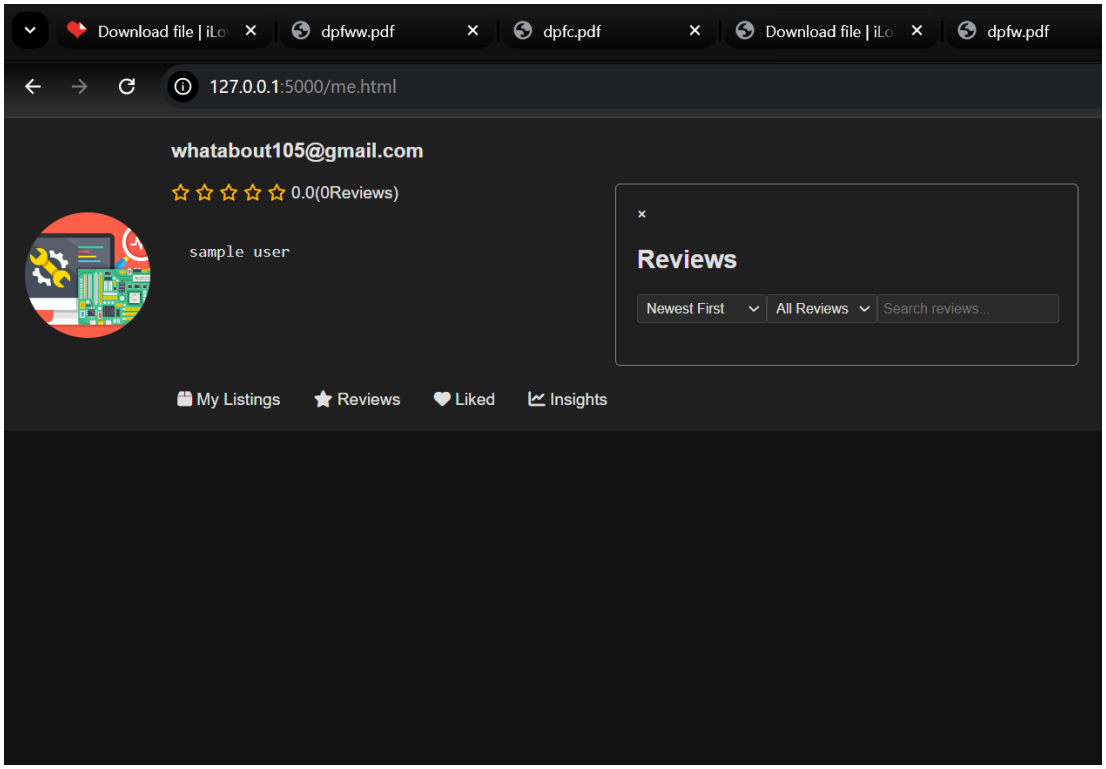

Figure 2.8: User Page

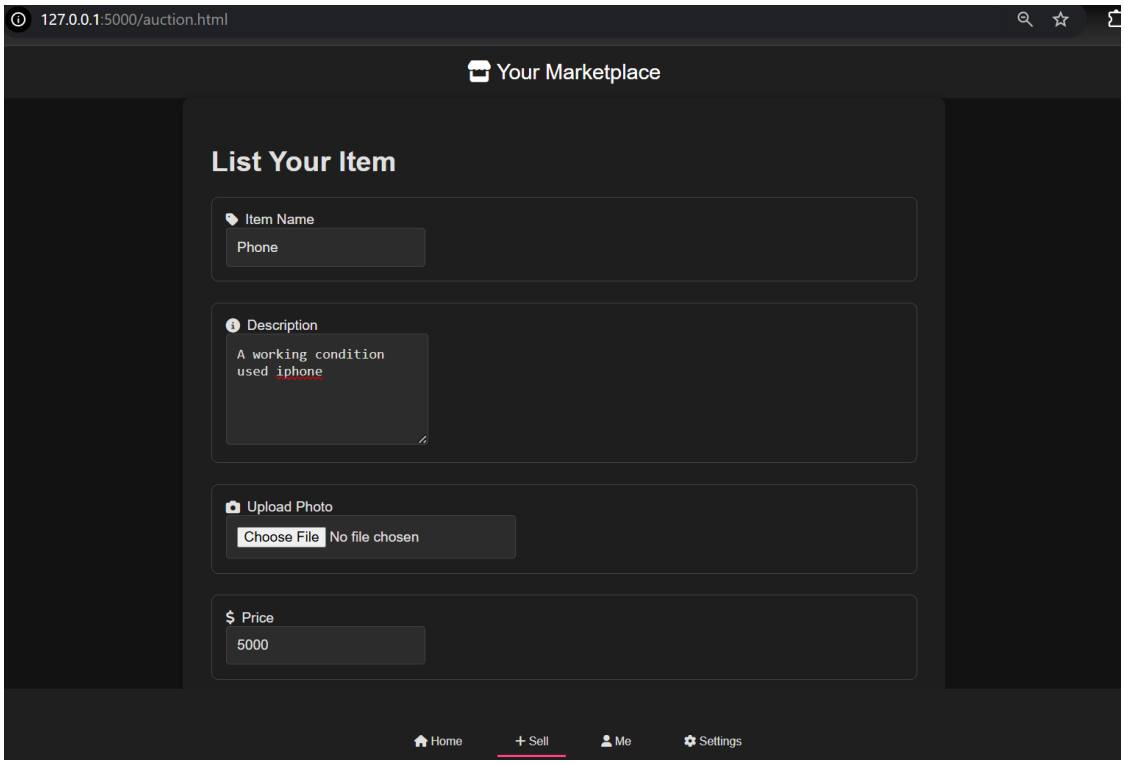

Figure 2.9: Listing

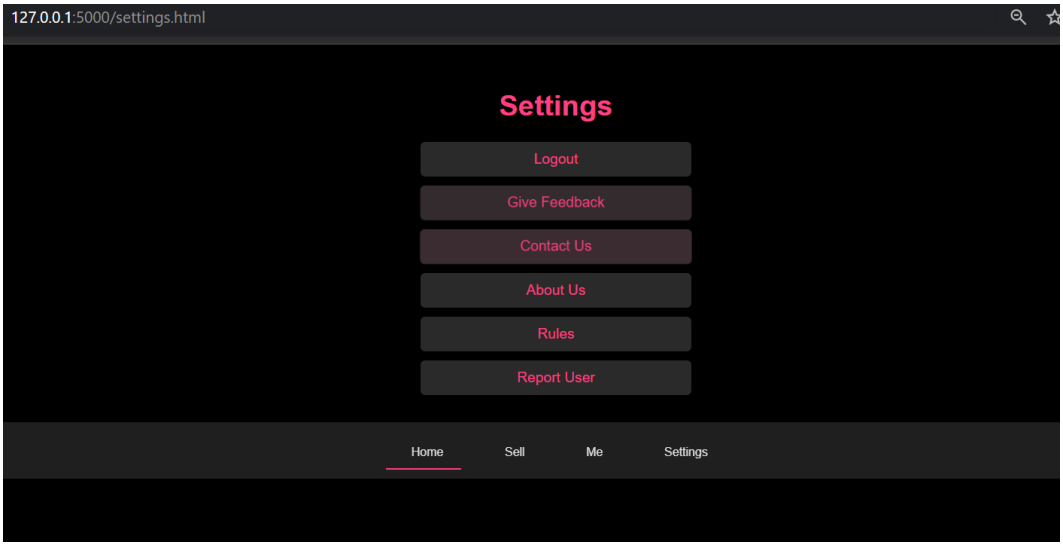

Figure 2.10: Settings Page

REFERENCES

1. Y. F. Lim and K. Hu, Online Business and Marketplaces, vol. 1, pp. 1–350, 2024.
2. T. Raja P. A., M. M. Iqbal, U. S. Umakanth, and A. Raj, Analysis of Online Marketplace for Local Vendors, vol. 5, no. 2, pp. 34–49, 2023.
3. A. Kumar, P. Sikdar, and R. Saha, Seller Experience Assessment in Online Marketplace, vol. 12, no. 4, pp. 223–240, 2021.
4. N. Peña-García, M. Losada-Otálora, and D. Pérez Auza, Reviews, Trust, and Customer Experience in Online Marketplaces, vol. 15, no. 1, pp. 67–89, 2024.
5. M. Luca, Designing Online Marketplaces: Trust and Reputation, vol. 9, no. 3, pp. 101–122, 2017.
6. G. J. P. S. Ferreira, D. P. Soares, and E. B. Silva, "Real-Time Messaging Systems for Online Marketplaces: A Case Study," IEEE Transactions on Mobile Computing, vol. 14, no. 8, pp. 1612–1625, 2015.
7. A. F. Yang, "Design and Architecture of an Online Marketplace: A Case Study," Journal of Computer Science and Technology, vol. 21, no. 4, pp. 512–522, 2006.
8. Y. Li, X. Yu, and Z. Wei, "The Impact of User Reviews and Feedback on E-commerce Purchases," Journal of Marketing Research, vol. 52, no. 5, pp. 342–356, 2015.
9. M. D. Zohdy and T. A. Majid, "Security Considerations in Online Marketplaces," IEEE Access, vol. 7, pp. 103195–103202, 2019.
10. A. B. Sharma and S. K. Sharma, "The Role of AI in Enhancing User Experience in E-Commerce Platforms," International Journal of Computer Science and Information Security, vol. 18, no. 2, pp. 98–112, 2022.
11. M. S. George, R. B. Harrison, and C. T. Williams, "Blockchain Technology for Secure Transactions in Online Marketplaces," Journal of Digital Commerce, vol. 13, no. 1, pp. 45–63, 2021.
12. J. M. Patel, "User Trust and Transaction Security in Online Marketplaces: A Case Study of Amazon," Journal of Business & Technology, vol. 30, no. 5, pp. 204–217, 2020.

13. L. T. Zhang, J. Li, and C. Zhang, "Optimization of Product Search and Recommendation Algorithms in E-Commerce Platforms," International Journal of Electronic Commerce, vol. 22, no. 4, pp. 135–153, 2018.
14. F. A. Dorsey, J. G. Phillips, and T. H. Moore, "Impact of Digital Marketplaces on Small Business Growth: Evidence from Emerging Economies," Journal of Business Research, vol. 68, no. 11, pp. 2456–2465, 2020.
15. R. P. Subramanian, "Mobile App Development for Online Marketplaces: Challenges and Innovations," International Journal of Software Engineering, vol. 41, no. 6, pp. 980–993, 2022.
16. S. K. Patel and R. M. Kumar, "Evaluating Seller Feedback Systems in E-Commerce Platforms," Journal of Marketing Technology, vol. 25, no. 3, pp. 142–157, 2023.
17. A. K. Soni, "The Future of Online Marketplaces: Trends, Technologies, and Transformations," Journal of Internet Commerce, vol. 16, no. 2, pp. 50–69, 2022.
18. J. L. Kim and S. H. Lee, "AI-Driven Personalization in E-Commerce: Enhancing User Experience through Data Analytics," Journal of E-Commerce and Business, vol. 17, no. 4, pp. 234–245, 2023.
19. T. G. Davies and P. J. Martin, "Machine Learning in Online Marketplaces: Improving Product Search and User Engagement," International Journal of E-Commerce Technology, vol. 19, no. 3, pp. 121–135, 2021.
20. V. R. Sinha and R. N. Gupta, "Fraud Prevention and Detection in Digital Marketplaces: A Machine Learning Approach," Journal of Cybersecurity and Digital Innovation, vol. 8, no. 1, pp. 98–113, 2022.