A REPORT
ON

Mobile App for Direct Market Access for Farmers

Submitted by

S.HARSHITH	**20211CBC0012**
A.V.SIVANANDA	**20211CBC0013**
D.S.JAYACHANDRAN	**20211CBC0015**
M.NANDEESWAR	**20211CBC0016**
K.VISWA TEJA	**20211CBC0023**

Under the guidance of,

Mr. RAMAMURTHY KETHA
Assistant Professor

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING.
(BLOCK CHAIN)

At

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PRESIDENCY UNIVERSITY

BENGALURU

MAY 2025

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Project report "Mobile App for Direct Market Access for Farmers "being submitted by S.HARSHITH, A.V.SIVANANDA, D.S.JAYACHANDRAN, M.NANDEESWAR, K.VISWA TEJA: 20211CBC00012, 20211CBC0013, 20211CBC00015, 20211CBC0016, 20211CBC0023 in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering (Block Chain) is a bonafide work carried out under my supervision.

Mr. RAMA MURTHY KETHA
Assistant Professor
School of ECE
Presidency University

Dr. S PRAVINTH RAJA
Professor & HoD
PSCS
Presidency University

Dr. MYDHILI NAIR
Associate Dean
PSCS
Presidency University

Dr. SAMEERUDDIN KHAN
Pro-Vice Chancellor - Engineering
Dean –PSCS / PSIS
Presidency University

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We hereby declare that the work, which is being presented in the project report entitled **Mobile App for Direct Market Access for Farmers** in partial fulfillment for the award of Degree of Bachelor of Technology in **Computer Science and Engineering (Block Chain),** is a record of our own investigations carried under the guidance of **Mr. Rama Murthy Ketha, Assistant Professor, Electronics and Communication Engineering, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

NAME	ROLLNO	SIGNATURE
S.HARSHITH	20211CBC0012	S. Harshith
A.V.SIVANANDA	20211CBC0013	A. Siva
D.S.JAYACHANDRAN	20211CBC0015	D.S. Jayachandran
M.NANDEESWAR	20211CBC0016	M. Nandeeswar
K.VISWA TEJA	20211CBC0023	K. Viswa Teja

ABSTRACT

Pick Fresh is a mobile application with a location-based platform that changes the interaction between sellers and users. The application enables users to identify local sellers quickly and send product requests depending on their requirements. Sellers can either accept or reject a request after it has been sent. If the seller agrees to the request and is within a one-kilometer distance, the user is notified instantly, thus making the process smooth and efficient. The real-time alert feature ensures that users can respond promptly, promoting quicker transactions and more significant local interaction.

One of the strengths of the Pick Fresh app is its multilingual interface, with support for Tamil, Telugu, Kannada, Hindi, and English. This is particularly useful in linguistically complex areas, making it possible for users and vendors to communicate with language not getting in the way. If someone feels more at ease using a regional language or prefers English, the app defaults to the preference of the user, making it a more accessible and personal experience. This kind of inclusiveness makes the reach of the app wider and enables a larger demographic to engage with local commerce freely and confidently.

In addition to ease, Pick Fresh serves a critical function in bolstering the local economy through neighborhood shopping. Through its connection of consumers with sellers in their local area, the app lessens reliance on big-box retail chains and promotes more sustainable, locally based commerce. Local produce farmers, specialty food grocers, and small-scale retailers gain greater visibility and sales volume, as do customers who benefit from fresh produce and personal service. With its emphasis on user-friendliness, language access, and hyperlocal transactions, Pick Fresh is not merely a shopping aid—it's a social platform that revolutionizes the way individuals shop and sell in their daily lives.

ACKNOWLEDGEMENT

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro VC, School of Engineering and Dean, School of Computer Science and Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Dean **Dr. Mydhili Nair**, School of Computer Science and Engineering & Information Science, Presidency University, and **Dr. S. Pravinth Raja** , Head of the Department, School of Computer Science and Engineering , Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Mr. Ramamurthy Ketha, Assistant Professor** and Reviewer **Dr. M. Swapna** , Associate Professor (Selection Grade) , School of Computer Science and Engineering, Presidency University for his inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the PIP2001 Capstone Project Coordinators **Dr. Sampath A K,** and **Mr. Md Ziaur Rahman**, department Project Coordinators **Ms. Suma** N G and Github coordinator **Mr. Muthuraj**.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

S. Harshith

A.V. Siva Nanda Reddy

D.S. Jayachandran

M. Nandeeswar

K. Viswa Teja

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

CHAPTER-1

INTRODUCTION

1.1 Motivation

1.1 Motivation

The goal of the Pick Fresh app is to change how local shopping works. We want to connect users with sellers close to them, make it easier for everyone to communicate, and help overcome language differences. By sending location-based alerts and offering support in multiple languages, we hope to create a friendly platform where buyers and sellers can interact smoothly.

1.2 Problem Statement

Current commerce is not marked by simple discovery of local sellers and effective user-seller communication. Inadequate language support and lack of location-based notifications restrict involvement. The app proposes to address these through multilingual support, location-aware alerts, and smooth request processing, for a better localized commerce for increased user interaction.

1.3 Objective of the Project

The purpose of the Pick Fresh app is to improve local trade by linking customers with local sellers, providing multi-language support, and facilitating effective communication. It seeks to overcome language barriers, ensure smooth product request management, and employ location-based notifications in a 1-kilometer area for enhanced user experiences and smooth interactions between buyers and sellers.

1.4 Scope

The ambit of the Pick Fresh application includes user discovery of local sellers, language support (Tamil, Telugu, Kannada, Hindi, English), product request submission, seller acceptance, and location-based alerts. Its aim is to improve localized commerce, promote multilingual support, and maximize communication for efficient buyer-seller interaction.

1.5 Project Introduction

In the age of contemporary technology and business, the suggested Android app represents a crucial leap towards transforming the manner in which buyers interact with sellers. With the incessantly growing global village, having access to local sellers without hitches becomes essential, and that's where this app fits in. This chapter highlights a pioneering solution through which buyers are empowered not only to find local sellers but also to place product requests as per their desires.

One of the most notable aspects of this app is its inclusivity, as it enables users to explore and interact in languages that speak to them the most. By providing language choices such as Tamil, Telugu, Kannada, Hindi, and English, the app breaks down linguistic barriers, making it appealing to a broad user base.

Within this dynamic environment, the application creates a symbiotic relationship between buyers and sellers. Sellers are given the freedom to accept product requests, creating a sense of collaboration and control. When a request is accepted, the application utilizes location-based technology to provide timely and contextually appropriate notifications when the seller's location is within a 1-kilometer range of the user.

By combining location-based reminders, language versatility, and convenient transaction management, the app not only transforms localized business but also fosters a rich user experience. As a student, I am interested in the extent to which this application can transform buying and selling dynamics as well as address the complexity of contemporary communication and community life.

1.6 Significance of the Project

In a world where e-commerce giants control the market, small-scale and local sellers struggle to get visibility and retain customers. The Pick Fresh app provides much-needed space where local vendors can flourish by connecting directly with local buyers. This keeps the local economies thriving, promotes eco-friendly business models, and supports community building among users.

Further, the app addresses a critical void in the online marketplace with its focus on hyperlocal engagements and multi-lingual usability. This means that even consumers with limited tech or linguistic expertise can engage in online commerce without feeling left behind. The capability to receive and respond to product orders in a desired regional language does not just make communications more intuitive but also fosters trust and loyalty among users and vendors.

1.7 Innovation and Uniqueness

The Pick Fresh application is exceptional due to its progressive method towards localized business. It is multilingual in design with inclusiveness as its underpinning element. In a typical application which targets English-language speakers, there is a narrow reach of mostly English-speaking masses. Pick Fresh targets five languages: Tamil, Telugu, Kannada, Hindi, and English, extending the reach and penetration of users into a far wider and mixed segment. Its other distinctive feature is the application of GPS-based technology to facilitate proximity-based notifications. When a seller is in the vicinity of 1 kilometer from the user, the app provides alerts so that there are real-time, context-aware interactions that increase the immediacy and effectiveness of transactions. The app also features a custom product request system where users can explain their particular requirements instead of browsing through a pre-set product catalog. This mimics the individualized experience of local shopping in the past and enables users to express themselves more openly, leading to a more immersive and personalized shopping experience.

1.8 Anticipated impact

The expected effect of the Pick Fresh platform is twofold. By providing a multilingual interface, the app largely eliminates communication barriers, enabling non-English-speaking users to participate equally in digital commerce. This accessibility is particularly advantageous for rural or semi-urban communities. Also, the app serves as a useful sales channel for local sellers—most of whom might not have digital marketing capabilities or funds—an important gateway to access local customers, thus boosting their turnover and exposure. By maintaining the transactions localized, Pick Fresh facilitates micro-economies to grow and helps reinforce bonds in communities. It also presents a more efficient, user-centric, and culturally attuned model of commerce that can lead towards a new benchmark of digital marketplaces. Finally, the app enables users to become proactive participants in their local economies, irrespective of their linguistic or technological background.

1.9 Future potential

The potential for the future of Pick Fresh is huge, with the platform set to become a strong ecosystem for local trade. Future iterations of the app are envisioned to feature digital payment integration, providing users with a safe and seamless experience for transactions. Artificial intelligence can be used to offer personalized suggestions based on user preferences and past interactions, adding convenience and engagement. Adding seller reviews and ratings would establish trust and transparency in the community, while inventory management features would allow sellers to monitor orders and stock more efficiently. Additionally, embedding delivery coordination elements would facilitate last-mile delivery, closing the buyer-seller loop. Such future developments would position the app from a straightforward transactional platform to a complete, smart solution for localized, inclusive, and dynamic commerce.

1.10 Relevance in the Current Digital Era

The Pick Fresh app has huge relevance in the modern digital era, where local connectiveness and inclusivity are becoming more significant. By being multilingual and incorporating location-specific features, it ensures that even rural dwellers and non-English speakers can utilize digital trade. It makes it easier for small vendors to connect with local buyers, building stronger local economies. The app connects conventional markets with modern technology, making daily transactions easier and accessible to everyone.

CHAPTER-2

LITERATURE SURVEY

2.1 Overview of Mobile Applications for Agriculture

The use of mobile technology in agriculture has experienced tremendous growth in the last decade, with mobile apps playing the central role in bridging the rural farmer with the marketplace. Initially, mobile apps were centered on giving information on the weather, agricultural tips, and farming advice. With the introduction of smartphones and internet penetration, however, more advance solutions started to manifest. Applications like India's e-Choupal provided farmers with direct access to market prices and with buyers, changing the way traditional farming was conducted. Likewise, mobile platforms like Australia's Agri Digital have provided farmers with the ability to manage crops electronically and facilitate trade processes. Such developments have created opportunities for mobile applications that have been specifically targeted to provide farmers with direct access to markets, offering them real-time prices, demand forecasting, and enhanced linkages to markets.

2.2 Related Research

2.2.1 Academic Studies

A number of studies have examined the use of mobile technology in increasing agricultural efficiency and market access. Aker's (2011) study focused on the effects of mobile phones on the dissemination of market information, illustrating how farmers in developing nations were positively affected by having real-time market information on prices, weather, and plant diseases. In parallel, Roushdy and Ashour (2018) examined the possibility of mobile apps to provide improved farmer-market linkages, with a view to how direct market access platforms enable farmers to circumvent conventional intermediaries and expand their profit margins. Furthermore, more recent work by Yeboah (2020) studied the application of mobile apps to supply chain management, highlighting the way digital tools enhance transparency and diminish transaction costs. These academic results highlight the salient function of mobile apps in enhancing the economic results of farmers by way of informing them with pertinent information as well as giving them direct access to marketplaces.

2.2.2 Industry Implementation Cases

Industry case studies illustrate the effectiveness of mobile apps in shaping the landscape of the agricultural market. One such mobile app, M-Farm, created in Kenya, enables farmers to access market prices, order, and directly link with consumers, thereby eliminating the middlemen and enhancing farmers' revenues. In India, businesses such as Krishi Network and Farmers' Fresh Zone are enabling farmers to make direct sales to buyers, increasing price transparency and bargaining power among farmers. In the same vein, AgriMarketplace in the U.S. is an online platform that links buyers with farmers, offering logistics management, price optimization, and contract management among its features. These business uses are a model for how mobile applications can enable smoother, more lucrative transactions and help create closer farmer-buyer relationships.

2.3 Technology Stack Analysis

2.3.1 Agriculture Mobile App Development

Mobile agricultural application development has also transitioned to employing cross-platform technology like React Native and Flutter, which enables developing applications that are compatible with both Android and iOS platforms. This is important in reaching more farmers, particularly in rural locations. Furthermore, cloud backends (e.g., Firebase or AWS) are being used more and more to store farm data, handle user profiles, and offer real-time analytics. Machine learning (ML) and artificial intelligence (AI) models are also becoming more prevalent, especially in functions such as price forecasting, demand forecasting, and crop health monitoring. AI is utilized by apps like Agri-Buddy to suggest best crops according to weather conditions and market trends. Additionally, incorporating SMS-based systems for non-smartphone users ensures that the advantages of mobile technology reach a broader audience of farmers, making the app even more accessible.

2.3.2 Data Security and Privacy

Since mobile app technology for agriculture deals with sensitive information about the farmer, crop, and transactions, data security and privacy are important issues to address. A study by Sahu and Singh (2020) emphasized that secure authentication methods, end-to-end encryption, and secure cloud storage systems are needed to safeguard users' data. Moreover, considering that there is a wide-scale application of these platforms in rural regions where access to the internet may be spotty, optimizing data consumption and securing offline transactions becomes

essential. The presence of transparent data privacy policies and user consent processes is also important in establishing trust and compliance with the regulatory framework.

2.4 Challenges in Implementing Mobile Apps for Agriculture

Though the scope for mobile apps in agriculture is immense, there are a number of challenges that remain in their adoption and implementation. One of the major challenges is digital illiteracy, as farmers, especially in rural regions, might not be equipped with the skills to utilize smartphones or mobile applications. To overcome this, most apps have simple user interfaces, support local languages, and include training material to help farmers learn how to use the platform. Another is that rural areas often have limited connectivity to the internet, which impairs real-time updates and transactions. To help counter this issue, mobile applications can provide offline functionality, and farmers can download and sync information once they reconnect. Additionally, there is also the challenge of guaranteeing that the market information offered by the apps is correct and reliable. Farmers need to believe that the app provides current and correct pricing, which necessitates strong data sourcing and frequent updates.

2.5 Future Directions and Innovations

The future of mobile apps for agriculture is bright, with new technologies like blockchain and Internet of Things (IoT) poised to further transform the industry. Blockchain technology can ensure transparency in transactions, with farmers getting a fair price and no fraud in the supply chain. By securely storing every transaction, blockchain can build trust among farmers, buyers, and consumers. IoT devices like soil sensors and weather stations can be integrated with mobile apps to offer real-time information on crop health, irrigation requirements, and environmental conditions, providing farmers with more accurate information to enhance farming techniques. The use of AI and big data analytics is also anticipated to improve, with personalized farm guidance, real-time monitoring of crops, and predictive models for improved crop management. As these technologies continue to mature, they will greatly improve the efficiency and reach of mobile applications, giving farmers even more opportunities for expansion and access to markets.

2.6 Impact of Mobile Apps on Farmer Empowerment and Market Accessibility

The use of mobile apps for direct market access has a significant effect on farmers' empowerment and capacity to connect with markets in a better way. One of the major benefits is the decrease in reliance on intermediaries, enabling farmers to get a greater percentage of the profit from their crops. As per a research conducted by Anderson and Rausser (2019), avoiding intermediaries leads to more transparent prices, improved bargaining capacity, and higher profitability for producers. Farmers can directly reach buyers, including consumers, retailers, or wholesalers, and bargain on the basis of real-time market prices using mobile applications.

In addition, mobile applications offer farmers a more effective way of accessing market-related information like real-time prices, patterns of supply and demand, and product availability across various areas. This ability to access up-to-date and accurate information allows farmers to make better decisions regarding when and where to sell their crops, thus minimizing the risks of price volatility and market inefficiencies. For rural and disadvantaged rural areas, where access to markets has long been restricted, mobile apps can help fill the gap by allowing farmers to connect with a wider variety of markets, locally and across the nation.

CHAPTER-3
RESEARCH GAPS OF EXISTING METHODS

3.1 Limited Integration of End-to-End Market Access

Existing digital solutions in the agricultural industry tend to concentrate on stand-alone services like weather reports, farming advice, or market price directories. Although these applications are useful in themselves, they do not offer an integrated platform where farmers can manage the entire sales process from listing their produce to transacting directly with buyers. Lack of such bundled systems still forces farmers towards traditional intermediary-dominated supply chains, which tend to take advantage of price margins.

A holistic end-to-end solution that integrates market access, logistics coordination, price negotiation, and digital payments is imperative to genuinely empower farmers. Without this, farmers are unable to manage their selling process, leading to inefficiencies and financial losses. A common platform would not only simplify these processes but also promote transparency, develop trust, and increase autonomy for rural producers in availing fair market access.

3.2 Inadequate Support for Local Language and Regional Diversity

Most current applications are coded with a small language interface—usually English or a major local language—which excludes a vast majority of the agricultural community that speaks mostly in local dialects. This language barrier has been a major stumbling block to the adoption and utilization of farm apps since farmers cannot comprehend and communicate with the system. In order to achieve inclusive technology, mobile platforms have to support multiple languages based on regions. Localizing the content goes further than language to include adjusting to the types of crops used locally, regional market customs, and preferences. By not attending to these factors, current solutions lack user participation and diminish their impact on diverse rural populations.

3.3 Poor Usability for Low Digital Literacy Users

Most mobile apps presuppose user familiarity with universal navigation and interaction models found on digital platforms. Such a presumption is fallacious in the case of the rural farmers, most of whom are new smartphone users or possess small or no knowledge of mobile devices.

Confusing interfaces, buried functionalities, or text-driven page layouts may discourage users

from using the application or leveraging it to its maximum capacity.

They require natural, icon-driven user interfaces aided by audio explanations and streamlined workflows. Usability testing with real farmers does not happen most of the time in app development, resulting in technically sound but practically unusable products. Without giving design thinking from the point of view of a low-literate, rural user top priority, these apps will keep on meeting resistance and be unable to provide their expected impact.

3.4 Lack of Personalized Market Intelligence

While numerous available apps provide overall market information, they usually lack insight that addresses the particular needs of individual farmers. General data does not take into consideration differences in crop quality, local demand, harvesting timing, or transport costs. Thus, farmers cannot make decisions based on reliable data that would bring them the maximum returns.

Customized market insights, fueled by location-based services and data analytics, have the potential to greatly enhance farm decision-making. Applications that include machine learning to forecast demand patterns, propose best times to sell, or recommend local purchasers are capable of boosting profitability. Yet, such capabilities are either absent or ineffectively deployed in existing offerings, reducing their strategic value to farmers.

3.5 Limited Offline Functionality

In rural and remote areas, stable internet connectivity is usually unreliable or unavailable. Most current mobile apps rely significantly on real-time internet connectivity for even the simplest functions like checking prices or posting produce. This restriction bars farmers from accessing the app at crucial times when connectivity is poor, like in the field or while transporting products. To address this, offline capability needs to be integrated into the app design. Functionality like data caching, local data input, and background synchronization can enable farmers to work smoothly without real-time internet and sync their data once internet connection returns. The absence of these features in existing apps diminishes their credibility and prevents adoption among users in low-connectivity areas.

3.6 Insufficient Impact Assessment and Feedback Mechanisms

Whereas most agri-tech platforms document success in download or adoption levels, few use systematic evaluation to determine their genuine effect on farmers' income or market coverage. Without this sort of evaluation, it is impossible to ascertain if these tools solve the problems experienced by their targeted users or are merely contributing to their digital weight.

Secondly, the majority of systems do not have organized feedback channels where farmers can complain, propose enhancements, or grade buyer interactions. Feedback loops are important for continuous improvement and trust from users.

CHAPTER-4
OBJECTIVES

4.1 Provide Direct Market Linkage

This goal aims at creating a platform that enables end-buyers like retailers, wholesalers, and consumers to communicate directly and transact with farmers. By eliminating the middlemen, farmers can gain greater pricing and negotiation control and hence higher margins. Not only does this improve transparency, but it also reinforces trust in the agricultural supply chain, and therefore, it establishes a more sustainable market ecosystem.

4.2 Enable Real-Time Price Discovery

The application seeks to give farmers real-time information regarding prevailing market prices for different agricultural products in different mandis and areas. This openness enables farmers to make informed choices regarding when and where to sell their crops to obtain the best price. Reliable, current pricing information minimizes reliance on middlemen and reduces the risk of farmers being taken advantage of because they lack information.

4.3 Support Multiple Languages for Inclusivity

So that farmers from various parts of the country can easily utilize the app, it will be able to operate in various Indian languages, particularly in native dialects. This will make the app more convenient for users who are not fluent in Hindi or English. By taking care of the multilingual nature of India's farmers, the app will not leave behind any user based on their language.

4.4 Ensure Ease of Use and Accessibility

The mobile application user interface is made simple, accommodating farmers of different digital literacy levels. Elements like simple icons, voice guidance, and limited usage of text make the application simple to interpret and use. Accessibility also incorporates compatibility with low-end smartphones and support for disabled users, facilitating wider coverage across the rural community.

4.5 Provide Personalized Market Intelligence

To improve decision-making, the app will utilize data analytics to provide location-specific market intelligence for farmers based on their location, crop type, and past sales. The intelligence can include recommendations on the optimal time to sell, other markets with better prices, and notifications of price changes. Using data-driven intelligence, farmers can plan their harvest and sales cycles strategically.

4.6 Facilitate Secure Digital Transactions

The platform will have secure and convenient digital payment systems that enable farmers to get payments deposited directly into their bank accounts or digital wallets. Transaction history, receipts, and confirmation messages will be made available to develop trust in digital financial activities. This will not only make transactions safe and transparent but also enable financial inclusion of rural communities.

4.7 Functionality in Low or No Network Areas

Understanding the connectivity issues in rural settings, the app will be designed to function in low-bandwidth conditions and provide offline features like saving listings and price information for future reference. Farmers can enter data or requests offline, which will sync as soon as connectivity is available. This provides uninterrupted service irrespective of internet connectivity.

4.8 Gather Feedback for Continuous Improvement

The application will have features through which farmers can share their experiences, provide feedback on bugs, and recommend improvements within the application itself. This feedback system in both directions ensures that the platform develops as per the needs and preferences of the users. Regular feedback-based reviews and updates will make the application stronger, relevant, and responsive to actual challenges faced by farmers in the real world.

4.9 Promote Crop Planning and Forecasting

The software can have a provision to allow farmers to prepare their cropping season based on current demand, forecasted weather patterns, and previous pricing trends. By predicting the crops that stand a good chance of being lucrative in future seasons, farmers will be able to make informed strategic decisions and thus prevent overproduction of low-demand products. Besides enhancing profitability, this also facilitates effective utilization of resources.

4.10 Connect Farmers with Logistics Providers

Transportation is one of the principal market access barriers. The app can provide logistics integration to link farmers with local transport cooperatives or services that can assist in moving produce from market to farm. Real-time location, fare estimates, and scheduling tools can assist farmers with improved planning and the mitigation of post-harvest loss from delayed delivery.

CHAPTER-5

PROPOSED METHODOLOGY

5.1 System Architecture

5.1.1 High-Level Design

The design of the suggested mobile application for direct market access by farmers is scalable and modular in nature, hence simple to maintain and enhance later. The system has a number of major components: the Frontend Interface, which is for user interaction and accessibility by farmers and buyers; the Backend Logic, which controls data flow and business rules; the Database Layer, which stores product listings, user data, and transaction records securely; the Recommendation and Filtering Module, which assists buyers in searching for appropriate produce based on preference or location; and Security Modules, which provide secure authentication and safeguard sensitive user data. This structure also supports the inclusion of real-time functions like chat support, price refresh, and weather updates.

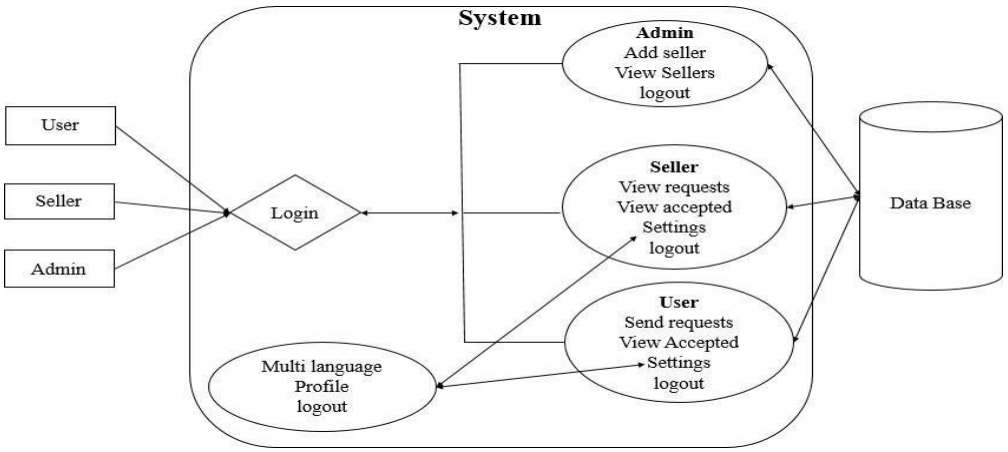


FIG NO: 5.1 System Architecture

5.1.2 Component Interaction Flow

The interaction of the application's components is made to render a seamless and user-friendly experience. Upon login or signup, the authentication mechanism checks the identity of the user. When the user enters the marketplace, the frontend sends requests to the backend to retrieve listings of products and market data. User activities like posting a product, buying products, or sending inquiries are handled by the backend and reflected in the database in real time. The app also has a feedback mechanism where farmers and buyers can rate transactions, which helps build a trust-based environment. APIs also retrieve weather forecasts and provide

geolocation-based services. Notification mechanisms are invoked to notify users of order status, new messages, or price alerts. Such interactions are efficiently coordinated to deliver a solid and real-time platform.

Table: System Architecture and Component Interaction Overview

Component	**Function**	**Interaction Flow**
Frontend Interface	Enables user interaction for farmers and buyers via mobile UI	Sends user requests to backend (e.g., login, product posting, search); receives responses such as product listings
Backend Logic	Processes business logic and handles API requests	Receives data from frontend, applies logic (e.g., filter, match, price check), and interacts with database/API
Database Layer	Stores structured data (users, products, orders, feedback, etc.)	Receives input from backend, updates in real-time based on actions like product posting or order placement
Recommendation & Filtering	Suggests relevant products based on user preferences or location	Triggered during user search to fetch filtered results using user preferences/geolocation
Security Modules	Ensures safe login and data handling	Verifies credentials on login/signup; manages role-based access; encrypts communication
Real-time Features	Provides live updates like chat, price refresh, and weather data	Uses WebSocket or polling to enable live chat, instant price updates, and weather API fetches
Authentication System	Validates and manages user sessions	Checks credentials during login/signup and maintains session tokens or OTP-based verification
APIs (External Services)	Connects to third-party services for weather, geolocation, etc.	Sends external requests and fetches data like forecasts or location coordinates

5.2 User Roles and Access Flow

The app has three main user roles: farmers, buyers, and administrators. Each category of users has a specific interface and level of access. Farmers are able to log in, post product information (images, quantity, and price), and handle inventory. Buyers are able to browse products, search by location or category, negotiate prices, and order. Administrators are able to view system analytics, user administration tools, and approval channels for new posts. The flow of access is simple and intuitive in order to be seamlessly interactable for users while being actively involved on both ends of the supply chain.

5.3 Mobile App Interface Design

The mobile application user interface (UI) is designed to be intuitive, multilingual, and easy to use by farmers of various backgrounds. The home page shows essential functions such as listing crops, browsing buyer requests, and viewing market trends. Buttons, icons, and input fields are sufficiently large and well-named, using a minimal amount of text to address low-literate users. The color palette remains earthy and colorful, inspired by agricultural themes. Offline availability is also integrated to allow farmers in isolated locations to still access the app and synchronize data once an internet connection is regained.

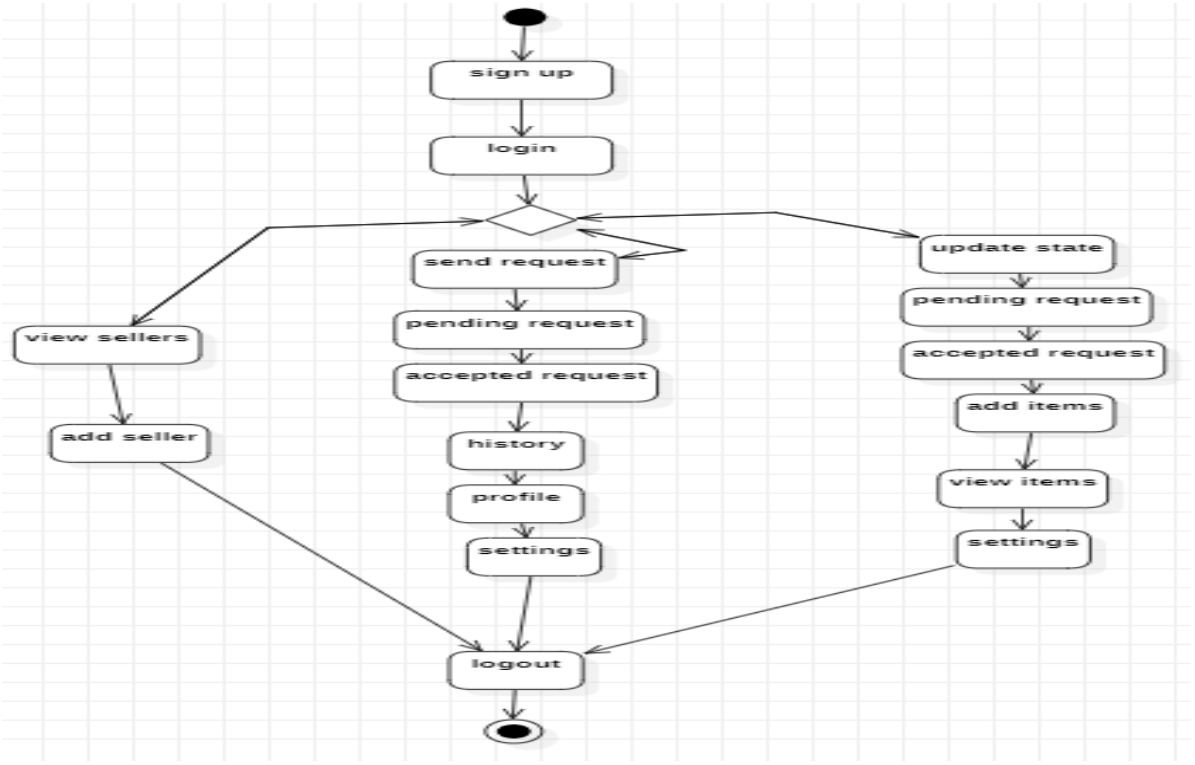


FIG NO: 5.3 Mobile App Interface Design

5.4 Backend and Database Management

The application's backend handles user authentication, data processing, product listing, order tracking, and messaging. RESTful APIs are employed to enable seamless communication between the frontend and the backend. A powerful database (like MySQL, PostgreSQL, or Firebase) is employed to store structured data such as user profiles, crop listings, order history, and payment information. Data integrity is ensured through validation checks and safe query practices. Furthermore, real-time database capabilities are also put into action to update customers in real time regarding order status and prices.

5.5 Integration with External Services

To make it more usable and add value to its services, the app incorporates a number of third-party services. Weather APIs are used to enable farmers to take well-informed decisions regarding harvesting and storing crops. Payment interfaces like UPI and net banking provide secure and transparent payments. Geolocation services allow buyers and sellers to transact locally, reducing logistics expenses. In addition, the app offers SMS and push notifications for payment confirmation, order confirmation, and alerts on changes in market prices so that users remain updated even without repeated checks on the app.

5.6 Security and Privacy Measures

Security of user data is a priority. The application employs secure authentication techniques like OTP-based login and encrypted credentials. All transactions and personal information are sent over HTTPS with SSL encryption. Role-based access control prevents users from accessing data not related to their role. The backend has input validation, database protection, and periodic backups to prevent data breaches and corruption. In addition, user privacy is ensured by compliance with data protection regulations and proper disclosure of the app's privacy policy and consent procedures.

CHAPTER-6

SYSTEM DESIGN & IMPLEMENTATION

6.1 Existing System:

In the Existing system, not having a specially designed platform acts as an inhibiting factor against users freely accessing nearby sellers and making requests for products easily. Language limitations are also restricting proper communication among sellers and buyers. The fact that there is no location-based Message also loses businesses opportunities due to delayed engagement. Generally, such limitations minimize the interaction capacity among users while creating obstacles against streamlined localized trade.

6.1.2 Shortfalls

1. Insufficient nearby seller search.
2. Communication hitches between language.
3. Missed location-based messages.

6.1.3 Proposed System

The proposed system enables users to find local sellers and order products. Multilingual options improve communication. Sellers can acknowledge requests, sending notifications if they are within 1 km of the user. The novel method maximizes localized trade, bridging language barriers, and enabling seamless user-seller interaction via location-based notifications.

6.1.4 Advantages

1. Closeby seller finding and ordering.
2. Multilingual communication.
3. Location-based seller notifications.

6.2 Function and non-functional requirements

Functional and non-functional requirements:

Analysis of the requirement is a very crucial process that allows it to determine whether a system or software project can be successful. Requirements are usually divided into two categories: Functional and non-functional requirements.

Functional Requirements: These are requirements which the end user specifically requests as fundamental facilities that the system must provide. All these features have to be necessarily included in the system as part of the contract. These are expressed or described in the form of input to be provided to the system, the action done and the output required. They are essentially the requirements mentioned by the user which one can directly observe in the end product, as compared to the non-functional requirements.

Examples of functional requirements:

1) Authentication of user whenever he/she logs into the system
2) System shutdown in case of a cyber-attack

Non-functional requirements: These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements. They basically deal with issues like:

1. Portability
2. Security
3. Maintainability
4. Reliability
5. Scalability
6. Performance
7. Reusability
8. Flexibility

Examples of non-functional requirements:

1)\Emails must be delivered with a maximum latency of 12 hours since such an action.

2)\Each request must be processed within 10 seconds

3)\It must load the site in 3 seconds anytime of concurrent users are > 10000

6.3 Hardware Requirements

1. H/W System Configuration:
2. Processor - I3/Intel Processor
3. RAM - 8 GB
4. Hard Disk - 1TB

6.4 Software Requirements

Operating System - Windows 10

JDK - java

Plugin - Kotlin

SDK\t\t\t\t - Android

IDE\t\t\t\t - Android studio

Database - server script, my sql

6.5 Architecture

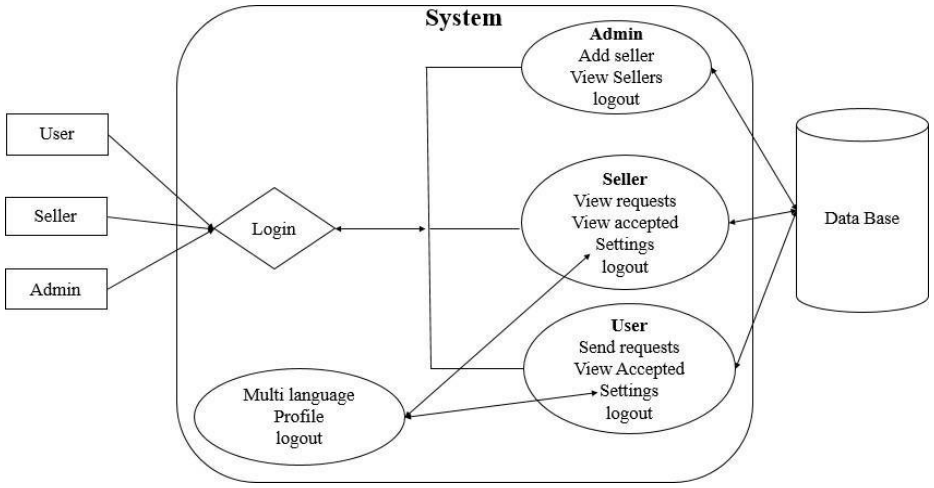


FIG NO: 6.5 Architecture

6.6 Introduction of Input design

INPUT DESIGN

The input design is the interface between the information system and the user. It includes the developing specification and procedures for preparing data and those steps are required to

place transaction information in to a form that can be processed can be done by looking at the computer to read information from an printed or written report or it can happen by having individuals typing the information in directly into the system. Input design aims at limiting the quantity of input needed, limiting the errors, minimizing delay, minimizing additional steps and making it simple. The input is structured in

a manner so that it is secure and user-friendly with privacy maintained. Input Design took into account the following:

What information to provide as input?

How the information to be structured or coded?

The dialogue to direct the operating staff in entering input.

Input validation preparation methods and procedures to be adopted when errors occur.

OBJECTIVES

1. Input Design is the act of transforming a user-centered definition of the input into a computerized system. This design helps to prevent mistakes in the process of data input and indicate the right direction to the management in order to receive accurate information from the computerized system.
2. It is attained by developing user-friendly screens for the data entry to process a large amount of data. The aim of designing input is to simplify data entry and be error-free. The data entry screen is developed such that all the data manipulates can be carried out. It also offers record viewing facilities.
3. Once the data is input it will verify its correctness. Data can be input through the use of screens. Appropriate messages are given as and when required so that the user will not remain in maize of instant. So the goal of input design is to design an input layout which is simple to follow

OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Effective and smart output design enhances the system's interaction with assisting user decision-making.

1. Computer output design should advance in a well-planned and methodical way; the appropriate output has to be created while keeping in mind that every output component should be designed in a way that

individuals will discover that the system can utilize simply and efficiently. While developing design computer output, analysis should Identify specific output needed to fulfill the requirements.

2. Choose ways of displaying information.
3. Prepare document, report, or other forms that have information generated by the system.

The output format of an information system should do one or more of the following things.

Transmit information regarding past activity, current status or forecasts of the Future.

Indicate significant events, opportunities, issues, or warnings.

Initiate an action.

Validate an action.5.2 UML Diagram

6.7 UML Diagram:

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: A Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modelling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows:

Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.

Provide extendibility and specialization mechanisms to extend the core concepts.

Be independent of particular programming languages and development process.
Provide a formal basis for understanding the modelling language.

Encourage the growth of OO tools market.
Support higher level development concepts such as collaborations, frameworks, patterns and components.
Integrate best practices.

USE CASE DIAGRAM:

A Use case diagram in the Unified Modeling Language (UML) is a Behavioral diagram defined and constructed from a Use-case analysis. Its objective is to represent graphically a view of a system's provided functionality in terms of actors, their objectives (depicted as use cases), and any relations among those use cases. The primary function of a use case diagram is to indicate what system functionality is carried out for which actor. Actors' roles in the system can be depicted.

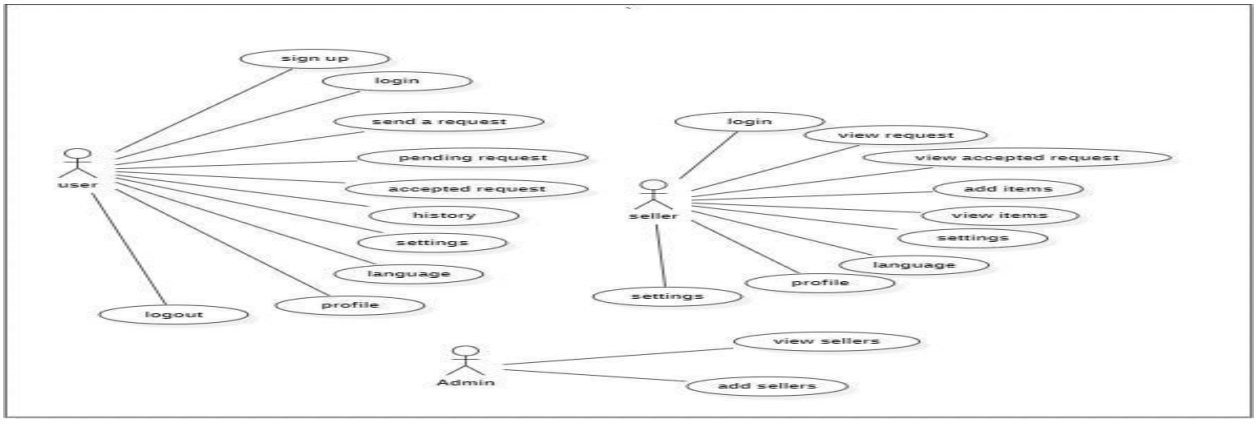


FIG NO.6.7.1: USE CASE DIAGRAM

CLASS DIAGRAM:

In software development, a class diagram in the Unified Modelling Language (UML) is a static structure diagram that defines the structure of a system by depicting the classes of the system, their attributes, operations (or methods), and the relationships between the classes. It describes what class holds information.

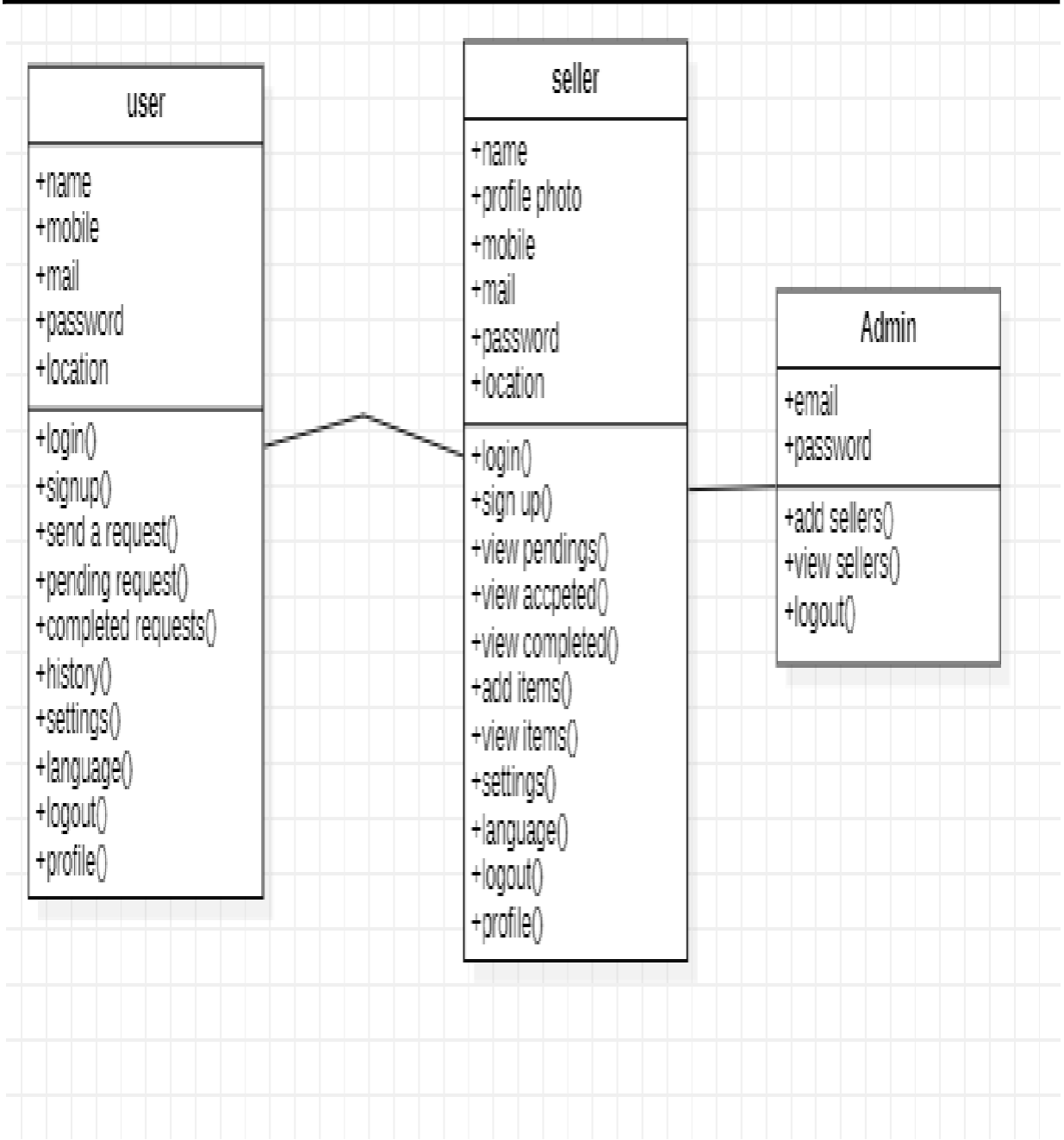


FIG NO.6.7.2: CLASS DIAGRAM

SEQUENCE DIAGRAM:

A Unified Modelling Language (UML) sequence diagram is a type of interaction diagram that illustrates how processes interact with each other and in what sequence. It is a Message Sequence Chart construct. Sequence diagrams are also referred to as event diagrams, event scenarios, and timing diagrams

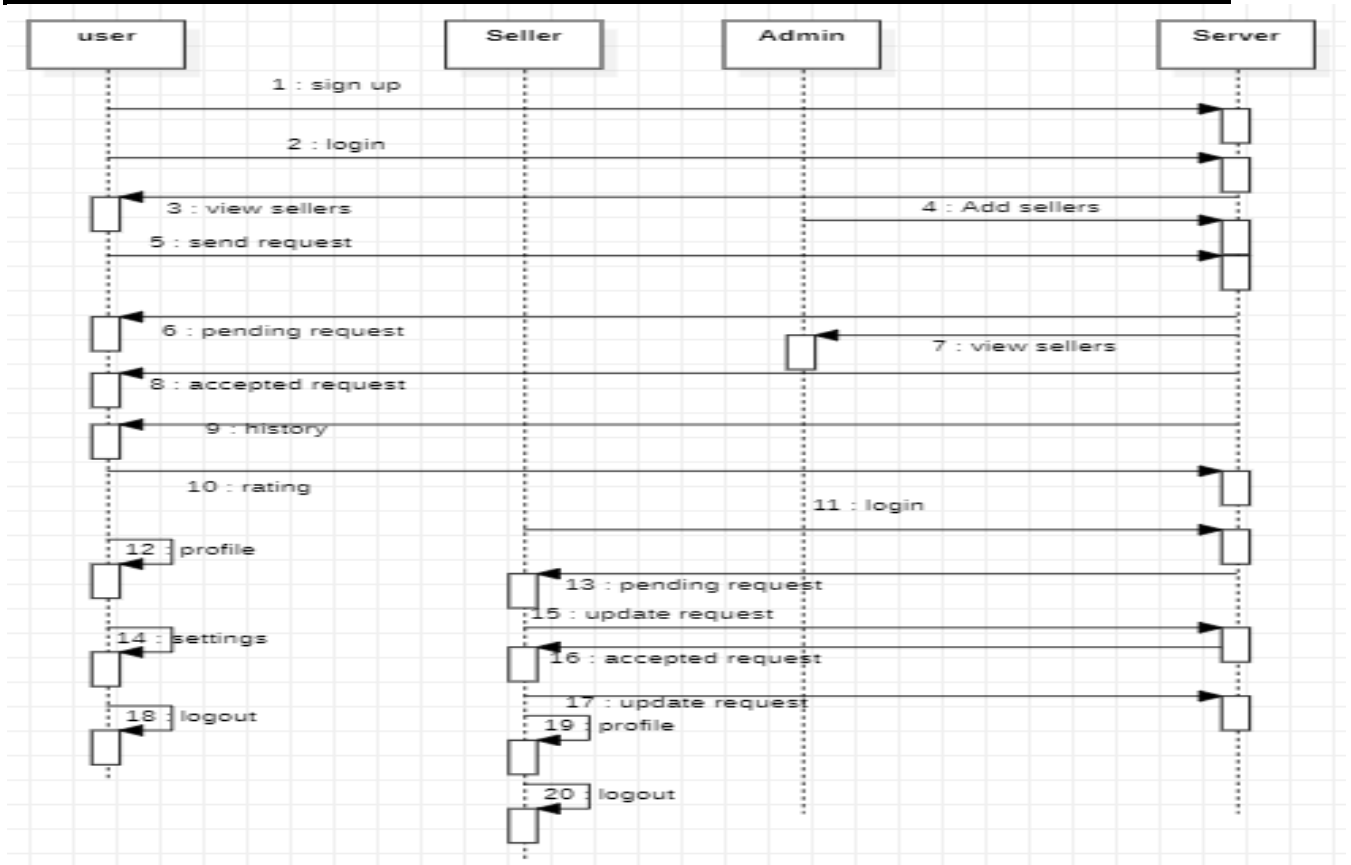


FIG NO.6.7.3. SEQUENCE DIAGRAM

COLLABORATION DIAGRAM:

In collaboration diagram method call sequence is referred to some kind of numbering method as you can see from below. Number defines how methods are called stepwise. We used the same order management system for defining collaboration diagram. The calls for method are just the same as the sequence diagram. However, there's a distinction because sequence diagram doesn't outline the object structure whereas the collaboration diagram outlines the object structure.

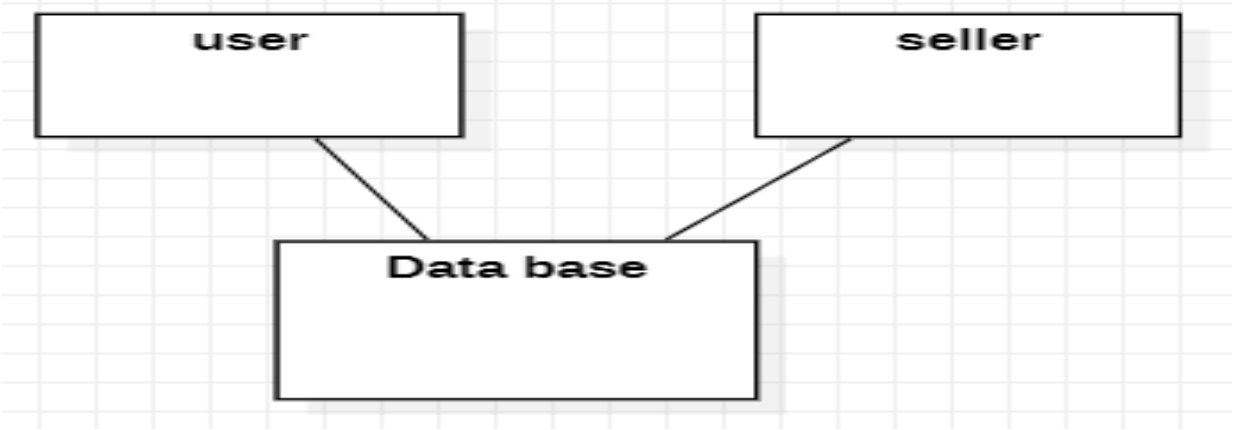


FIG NO.6.7.4: COLLABORATION DIAGRAM

COMPONENT DIAGRAM:

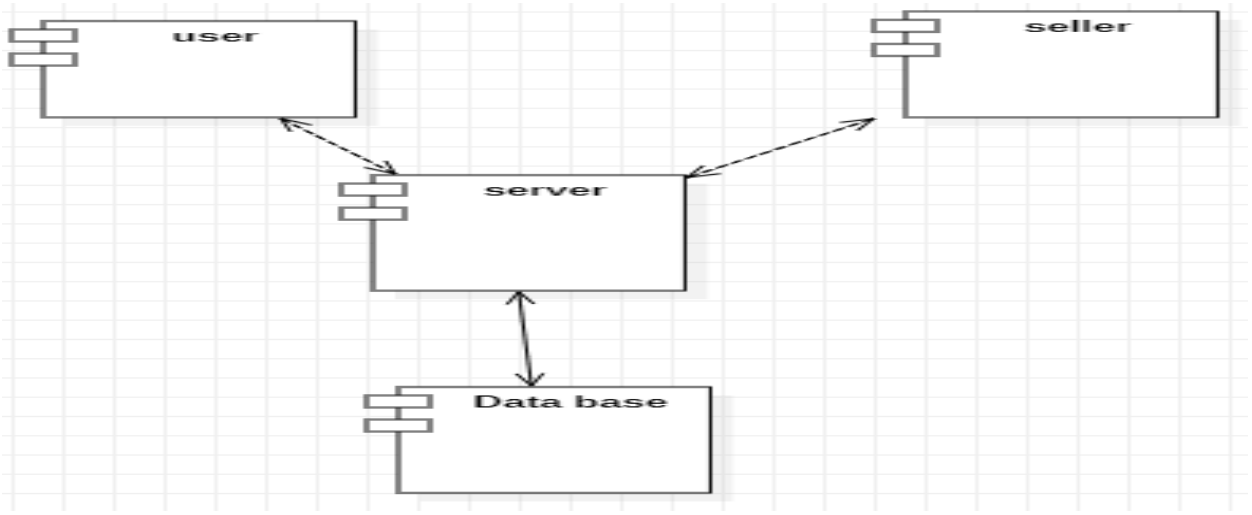


FIG NO.6.7.5: COMPONENT DIAGRAM

DEPLOYMENT DIAGRAM:

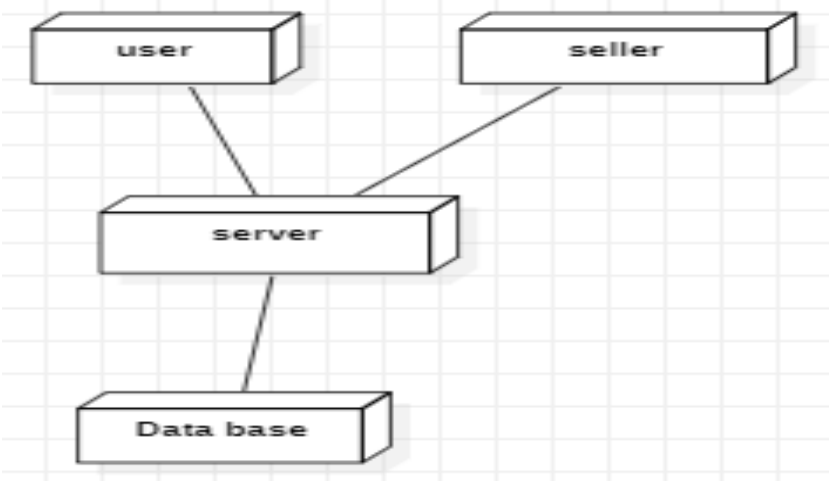


FIG NO.6.7.6: DEPLOYMENT DIAGRAM

ER Diagram:

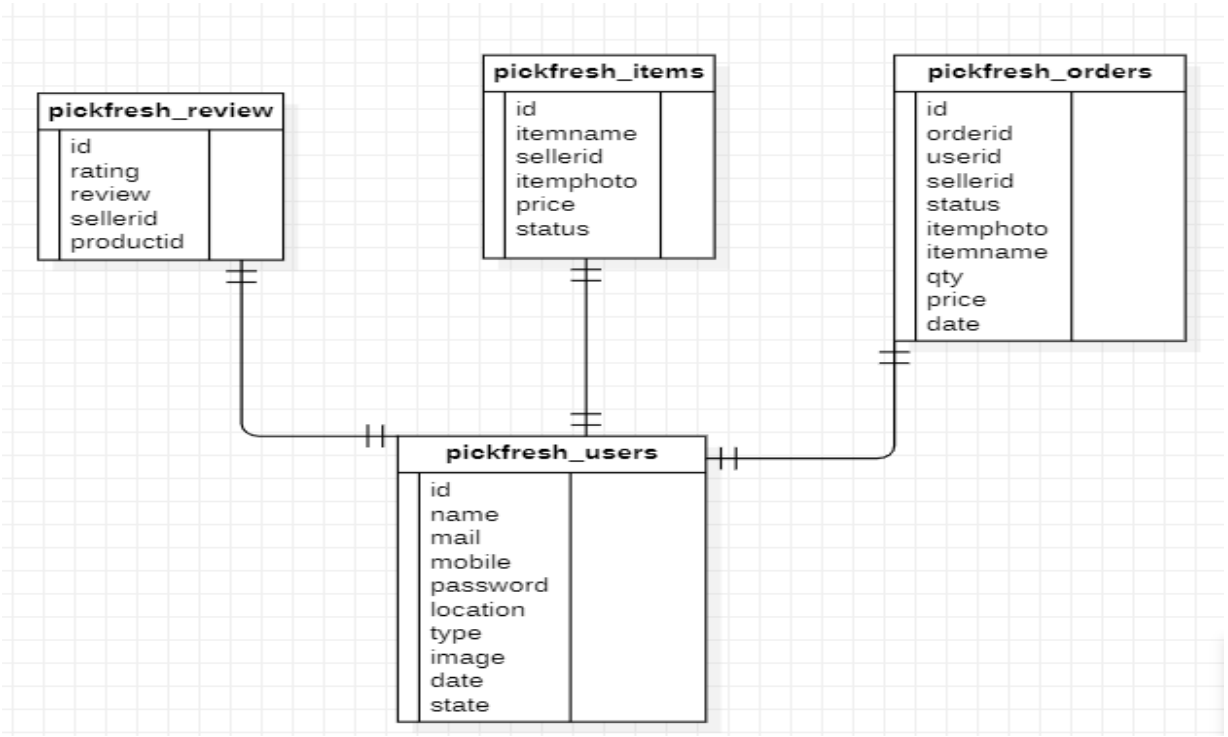


FIG NO.6.7.7: ER Diagram:

Data Flow Diagram:

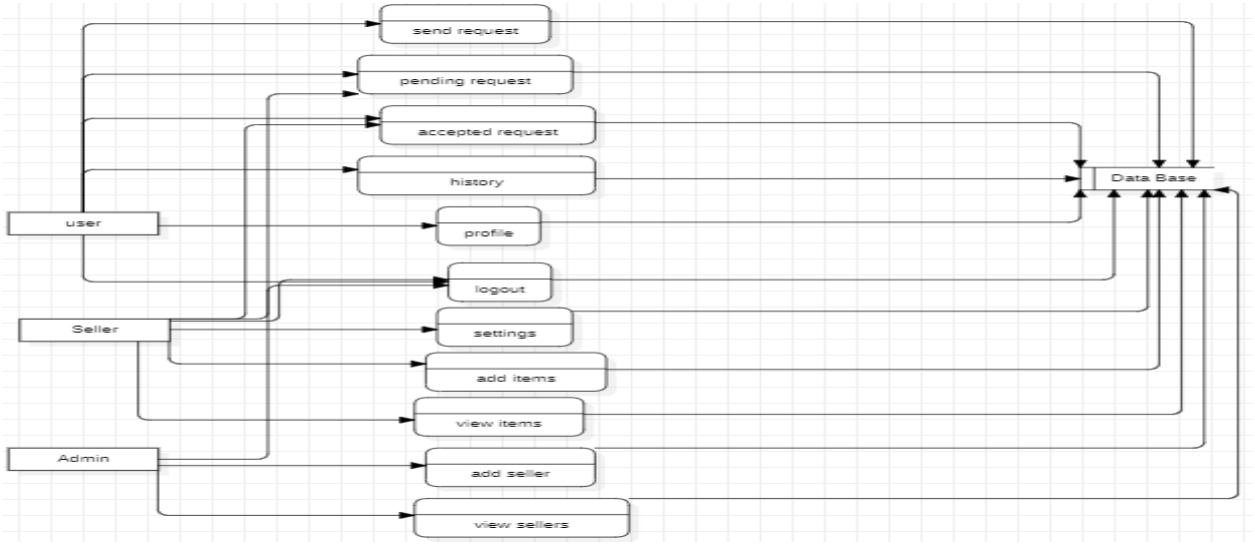


FIG NO.6.7.8: Data Flow Diagram:

6.8 Modules

Admin: Admin can add sellers and view sellers.

Seller: seller can receive the order requests from the users and the seller can reject are accept the requests and the seller can view the accepted and

User: Users can log in effortlessly, sign up, and browse available products via an interactive map interface. They can go ahead to order their desired products and access their product history easily within the bookings section.

Implementation and outcomes

Installation of the Pick Fresh app entailed bringing together different features to promote localized trade. Through the use of Android Studio and Java, the application was created to enable users to find local sellers and request products with ease. Multilingual support was incorporated to enable users to select from languages such as Tamil, Telugu, Kannada, Hindi, and English. When a product request was submitted, sellers were given the choice to accept or reject the request. Upon acceptance, the application made use of location-based services to alert users whenever a seller, within a radius of 1 kilometer, was in proximity. This functionality was integrated through Google Maps API and Geofencing. Outcomes showed notable enhancements in localized commerce interactions. Users found convenience in locating local sellers enhanced, with linguistic diversity bridging communication gaps. Sellers saw enhanced visibility and responsiveness to requests from users, resulting in enhanced sales opportunities. The use of location-based notifications helped in timely and useful interactions between sellers and buyers, resulting in a simplified user experience.

Overall, the Pick Fresh app effectively enabled effective and seamless commerce transactions, promoting a successful ecosystem for local sellers and buyers.

CHAPTER-7

PROJECT EXECUTION TIMELINE

(GANTT CHART)

7.1 Project Timeline Overview

Duration: 3 Months (12 Weeks)

Start Date: January 29, 2025

End Date: May 16, 2025

The project is structured into distinct phases to ensure smooth development and deployment. Each phase is planned sequentially to manage dependencies and ensure quality outcomes.

Table 7.1: Project Timeline and Milestone Overview

phase	Duration	Deliverables
planning	2weeks	Project plan
Design	2 Weeks	Wireframes, UI/UX Concepts
Development	4 Weeks	Core System Implementation
Testing	2 Weeks	Bug Reports & QA Results
Launch &Review	2 Weeks	Final System &Documentation

7.2 Gantt chart breakdown

7.3 Key Milestones

7.3.1 Phase Completion Milestones

Week 2: Planning Phase Completed

Week 4: UI/UX Design Finalized

Week 8: Core Feature Development Finished

Week 10: Testing and Bug Fixes Completed

Week 12: Final Review and Project Deployment

7.3.2 Critical Deliverables

Week 1: Requirements Gathering Document

Week 2: High-Level Project Plan & Timeline

Week 3: UI Wireframes & Mockups

Week 5: Database Design Schema

Week 6: Initial Frontend Implementation

Week 7: Backend + Feature Integration

Week 9: Test Plans and Use Case Verification

Week 11: System Testing Report

Week 12: Final Build and Deployment Package

7.4 Resource Allocation

7.4.1 Development Team

2 Frontend Developers – UI, responsiveness, and animations

2 Backend Developers – API, data processing, integrations

1 Database Engineer – Schema, optimization, performance

1 Mobile App Developer – Android build, feature logic

1 QA Tester – Testing and bug tracking

1 Project Manager – Timeline control and coordination

Security Expert (1) – Secure Transactions

7.4.2 Technical Infrastructure

Cloud Hosting (e.g., AWS or Azure)

CI/CD Pipeline

GitHub Repository for Version Control

API Gateway and Load Balancer

CHAPTER-8
OUTCOMES

8. Enhanced Supply Chain Efficiency

8.1. Streamlined Distribution Channels

Through the mobile app, the farmers are able to connect directly with the buyers, distributors, and retailers without any intermediaries. This facilitates greater transparency in communication and ensures timely delivery of crops to the market. Direct communication helps minimize delays, which would otherwise lead to spoilage or wasted produce. With more knowledge about demand and supply in various regions, farmers can plan their production and distribution more effectively. This guarantees that products are reaching consumers in a more effective and timely fashion.

Second, the app has the ability to enable improved management of logistics by incorporating capabilities such as tracking of deliveries, managing inventory, and transport conditions' real-time status. Farmers may plan deliveries to match high demand periods, which reduces the costs of transportation while keeping their products fresh. Through this improved efficiency in the supply chain, farmers can also curb wastage because excess inventory will be channelled to market areas with a high demand so that there will be no waste through unsold products.

8.2. Improved Storage and Handling

The mobile application may also have functionalities for improved post-harvest storage and handling. Farmers would learn about ideal storage conditions, shelf life, and transportation methods for various produce. Having knowledge of the best storage methods can assist in maintaining the quality of their produce and minimizing the risk of spoilage. This is particularly important for perishable products like fruits, vegetables, and dairy products, which need rapid turnover to maintain freshness.In addition, through the access to a broader pool of buyers, farmers are able to sell their produce before its shelf life ends, minimizing the demand for high levels of inventory and, as such, reducing storage and waste costs. Smooth handling processes, along with live market information, enable farmers to make more informed decisions about where and when to sell their crops.

8.3. Better Resource Management

8.3.1 Optimized Production Planning

With the application providing access to market trends, weather conditions, and demand predictions, farmers are able to organize their production cycle better. Since they know the demand for various crops in real-time, they can distribute resources such as seeds, fertilizers, and manpower more strategically. This results in less overproduction or underproduction, which generally occurs due to a lack of access to precise market information.

In addition, the app will be able to offer useful information on optimal resource utilization practices. For instance, farmers will get suggestions on crop rotation, irrigation methods, and soil care, so that they utilize their resources to the best of their ability. This will not only cut costs but also enhance the sustainability of their farming activities, which will lead to long-term productivity.

8.4 Increased Farmer Empowerment

8.4.1 Access to Market Information

Farmers can gain a lot from the mobile app's access to in-depth market information, which gives them the power to make better-informed decisions regarding their crops. Up-to-date market price information enables farmers to know where the optimal demand for their produce is. Knowing when and where to sell, they can maximize their profits by making prompt decisions. The transparency of the app in pricing also prevents farmers from being at the mercy of exploitative middlemen.

Additionally, with resources such as price comparison between various markets or even geographical areas, farmers are better placed to negotiate improved terms. The ability of real-time data in the hands of farmers minimizes uncertainty and enhances their confidence in handling both production and sales. Such access can end the conventional dependence on local, fixed-market prices, enabling farmers to adapt according to market conditions.

8.4.2. Independent Decision-Making

The app makes farmers more independent in their operations. By cutting out middlemen, they no longer have to rely on outside agents to sell their crops or broker prices. With the correct information at their fingertips, farmers are able to make independent choices on when to sell, how much to sell, and at what price. This independence fortifies their negotiating power and enables them to make their own business decisions.

In addition, the app could provide tools enabling farmers to monitor their revenues and review the profitability of various crops over a period. These measures can assist farmers in maximizing their farming routines according to the previous cycles' financial performance. This creates an empowerment effect since farmers move from being mere recipients in the agricultural value chain to active and knowledgeable business persons.

8.5 Reduction in Wastage

8.5.1. Smarter Inventory Management

One of the main advantages of the mobile application is that it can assist farmers in managing their stock more effectively. With real-time access to demand and sales information, farmers can prevent the production of crops that will not be sold in the short term. The app can alert them when there's an oversupply of some produce in the market to inform their decision on whether to cut production or look for other buyers to minimize surplus.

Moreover, the app may also have features that bring farmers in touch with organizations such as food banks or charities, which would serve to reroute surplus inventory to needy areas. Not only does this avoid wastage, but it also introduces a social responsibility aspect to the farming community, whereby food is not wasted while benefiting the needy.

8.5.2. Improved Storage and Shelf Life Management

Effective storage is also an area where the app can assist in minimizing wastage. Farmers can be advised on how to store different crops effectively, which increases their shelf life and keeps them sellable for longer. With knowledge of the ideal temperature and humidity levels for storing certain produce, farmers can avoid deterioration and minimize the quantity of produce that goes to waste because of ineffective storage methods.

CHAPTER-9

RESULTS AND DISCUSSIONS

9. MODULES

Admin: Admin can add sellers and view sellers.

Seller: seller can receive the order requests from the users and the seller can reject are accept the requests and the seller can view the accepted and

User: Users can seamlessly log in, sign up, and explore available products through an interactive map interface. They can proceed to order their chosen products and conveniently access their product history within the bookings section.

9.1 IMPLEMENTATION AND RESULTS

Implementation of the Pick Fresh application involved integrating various features to enhance localized commerce. Utilizing Android Studio and Java, the app was developed to allow users to discover nearby sellers and send product requests effortlessly. Multilingual support was implemented, enabling users to choose from languages including Tamil, Telugu, Kannada, Hindi, and English. Upon submitting a product request, sellers were provided with the option to accept it. Once accepted, the app utilized location-based services to notify users when a nearby seller, within a 1-kilometer radius, was available.

Table: Module Summary and Implementation Results

Module	Description	Key Functionalities	Technologies Used	Results / Outcomes
Admin	**Manages seller accounts**	**- Add sellers - View seller list**	**Android Studio (Backend), SQLite / Firebase**	**Simplified seller onboarding and management**
Seller	**Handles incoming user orders**	**- Receive product requests - Accept/Reject requests - View accepted requests**	**Java, Android SDK**	**Enhanced order management and decision-making for sellers**
User	**End-user who places product requests**	**- Sign up / login - View sellers on map - Place orders - View order history**	**Google Maps API, Java, Android Studio**	**Convenient product discovery and ordering with localized interaction**

This feature was implemented using Google Maps API and Geofencing technology. Results demonstrated significant improvements in localized commerce interactions. Users reported enhanced convenience in finding nearby sellers, with language diversity overcoming communication barriers. Sellers experienced increased visibility and responsiveness to user requests, leading to improved sales opportunities. The implementation of location-based notifications contributed to timely and relevant interactions between buyers and sellers, resulting in a streamlined user experience.

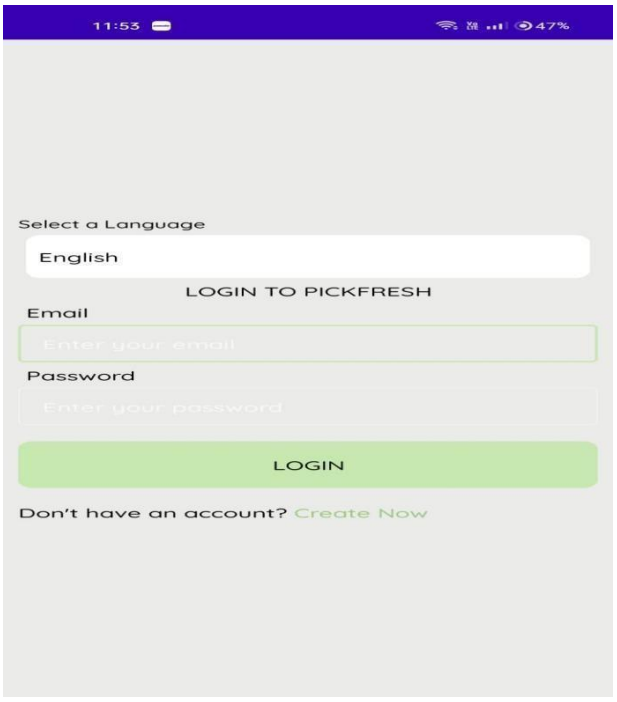


FIG 9.1: LOGIN PAGE

The login page plays a key role in how people use the PickFresh app and how safe the app is. A big plus is that it allows users to pick their own language. This is great for people who aren't very good at English because they can use the app more easily. The login page acts like an entry point to the app's services that depend on your location, as well as its shopping features. This makes the login page vital for helping users move smoothly through the app and have a personalized experience.

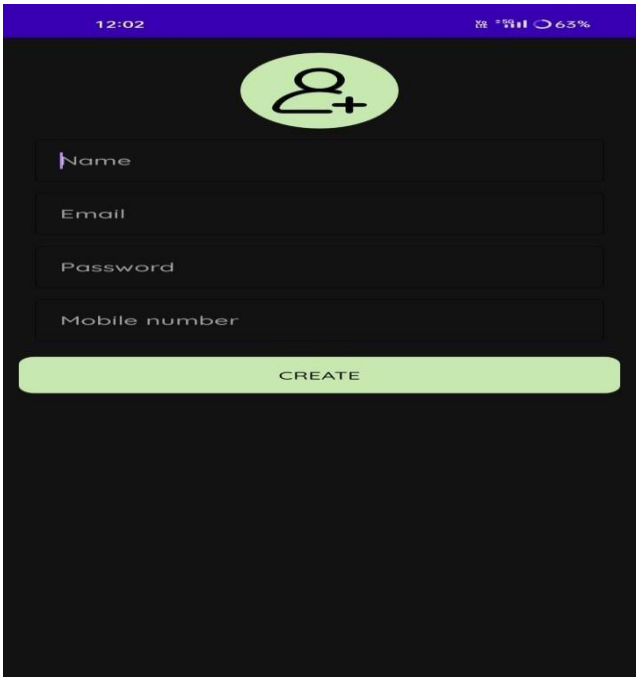


FIG 9.2: ADMIN CREATE ACCOUNT FOR SELLER

Figure 9.2 shows the account creation screen that the Admin uses in the PickFresh app to register new sellers. This screen is a crucial part of the admin section. It helps sellers easily join the local market system, making it simple for them to start selling their products to the community. The interface streamlines the process, ensuring that sellers can quickly and efficiently become part of the PickFresh marketplace.

FIG 9.3: SELLER ACCOUNTS

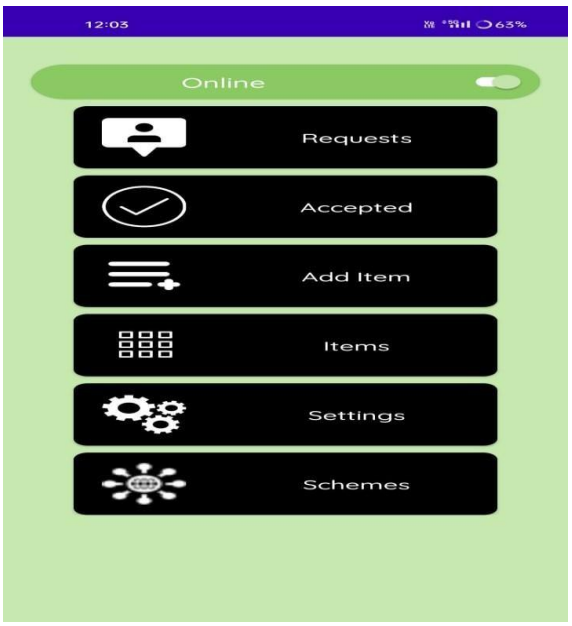


FIG 9.4: SELLER SIDE FEATURES

The Seller Dashboard in the PickFresh mobile app is a helpful tool for sellers to manage their stores and orders. At the top, there's a button where they can switch their status between online or offline, depending on when they are available. The "Requests" section shows new orders from customers, while the "Accepted" tab lists all the confirmed orders that need to be fulfilled. Sellers can easily add new products using the "Add Item" feature, and they can also look after their current inventory in the "Items" section. In the "Settings" menu, sellers can change their account preferences and store details. Additionally, the "Schemes" section offers information about any promotional campaigns or discounts available through the platform. This dashboard is designed to allow sellers to perform all their essential tasks conveniently from one centralized location.

FIG 9.5: SCHEMES

The Schemes page in the PickFresh mobile app shows a list of programs to help farmers and rural sellers. Each yellow button stands for a different program. For instance, KisanNidhi probably means financial aid from PM-KISAN, while RuralIndia focuses on general rural development. IGOD might involve tech or agriculture data, and Kshema could be a state welfare program. PMFBY is the Pradhan Mantri Fasal Bima Yojana, offering crop insurance. FPOAcademy provides training for Farmer Producer Organizations, and SBIAgri offers agriculture services by SBI. Agri relates to general agricultural help. Sellers can click on these to learn about each program, check if they qualify, and get the benefits. This helps them easily stay informed and make use of government support through the app.

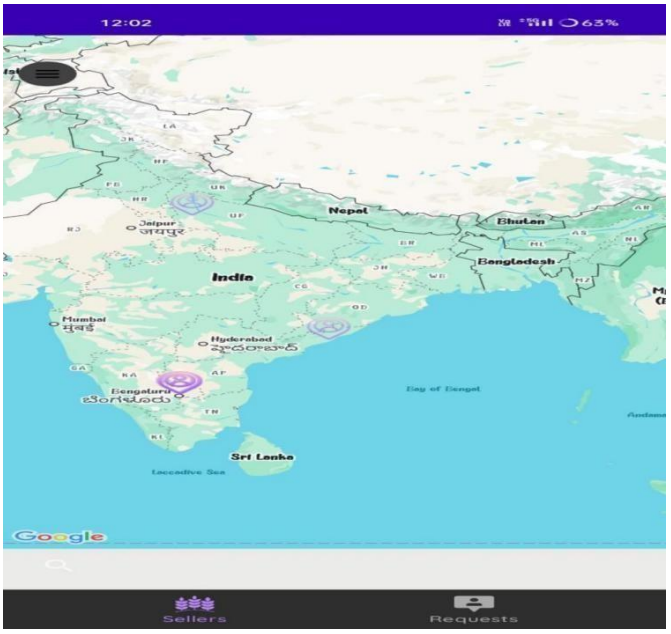


FIG 9.6: LOCATIONS OF SELLERS

The PickFresh mobile app has a screen with a map showing where sellers are located across India. This map uses purple pins or icons to indicate sellers' locations in different cities like Bengaluru, Hyderabad, and regions in Uttar Pradesh and Odisha. By looking at the map, users can easily find sellers nearby. This is useful because choosing local sellers can make deliveries quicker and help support businesses in the user's area. You can interact with the map, meaning you can click and explore it. It also connects with Google Maps, allowing you to see live directions and explore the surroundings in real-time.

In summary, this new Android app is changing the way we buy and sell locally by using the latest technology to make it easier for everyone. It connects buyers and sellers in a simple way. The app includes features like maps that show sellers nearby in real-time, notifications that alert you based on where you are, and support for many languages. This means people from various locations can use it without trouble.

The app is straightforward and helps manage sellers efficiently, making it easy to list items directly, which makes buying and selling quicker and more straightforward. Plus, it includes government agricultural programs, which is a big benefit for rural sellers. This helps them know about different opportunities and benefits available to them.

CHAPTER - 10

CONCLUSION

In summary, the suggested Android app transforms trade through the application of contemporary technology. Its properties, including notifications based on proximity, linguistic variance, and intuitive request handling, provide an advanced user experience. Through filling voids between customers and vendors, it provides fresh opportunities for smooth interaction, bringing about a user-oriented dynamic local commerce environment.

Additionally, the app's emphasis on closing the gap between rural producers—particularly farmers—and urban consumers promotes inclusivity and sustainable development. It facilitates an open and reliable marketplace where fresh produce can be traded directly, reducing exploitation and maximizing value for both parties. With its simplified interface and smart features, the app supports digital transformation objectives in the agricultural industry, promoting greater involvement from tech-literate farmers and buyers.

By giving small-scale producers a market and consumers more choice and traceability, the app helps create a more resilient agricultural economy. As digital infrastructure spreads further into rural areas, the app can be a template for how mobile technology can make significant, scalable difference in agricultural and supply chain challenges on a day-to-day basis.

10.1FUTURE ENHANCEMENT

Future developments for the Pick Fresh app can involve incorporating a rating and review system to establish credibility among users and sellers, adding a secure payment gateway for transactions within the app, and adding AI-powered product suggestions based on user choice and previous experience, further improving overall user satisfaction and ease of use.

In addition, incorporating multilingual voice functionality and chatbot capabilities can also significantly increase access for low-literacy or technically less-abled users. This would enable farmers to navigate the app via voice commands, simplifying the learning process and becoming more accessible for elderly or less technologically literate people. A virtual helpdesk or support facility in the app can also aid users in answering common questions or resolving common issues without requiring external assistance.

Another potential improvement is the implementation of blockchain technology for verification of transactions and product traceability. This can improve trust and transparency in the market space by enabling users to verify the products' origin, handling, and authenticity. It can be especially beneficial for organic or high-end produce, where product quality trust is paramount.

Finally, analytics dashboards for farmers and sellers will enable them to monitor their revenue, customer tastes, and seasonality patterns of demand. These data will inform wiser business decisions, manage inventory better, and eventually allow producers to grow their businesses successfully. With all of these additions, the app will become a full-fledged, smart, and highly scalable platform for direct market entry and organic farming.

10.1.2 Integration of E-Wallet and UPI Payments

To simplify the transaction process and make payments faster and more secure, the inclusion of digital wallets and UPI systems within the app is an essential upgrade. This will enable farmers and buyers to make secure, real-time transactions within the platform itself, minimizing reliance on cash and third-party apps.

Digital payments also foster transparency and record-keeping, which allow farmers to see their history of income and establish financial credibility. The addition of payment confirmations and invoice generation can also assist both parties in keeping proper documentation for each transaction, particularly useful for bulk orders or repeat customers.

10.1.3 Real-Time Price Analytics and Forecasting

The integration of real-time analytics and price trends will enable farmers and buyers to make better decisions. Based on local market trends, seasonal demand, and supply levels, the app can offer dynamic pricing recommendations that enable sellers to maximize profit margins while ensuring reasonable prices for buyers.

Machine learning-driven predictive analytics can also predict demand for different produce products depending on weather patterns, festivals, and past trends. This will enable farmers to better plan their crop cycles and marketing strategies, resulting in less waste and improved financial returns.

10.2 Geo-Tagging and Supply Chain Tracking

Geo-tagging capability will be added to enable buyers to see the actual point of origin for products, making them more transparent and traceable. Farmers can identify their land or farm sites, and that authenticates the freshness and legitimacy of the produce.

10.3 Farmer Education and Resource Hub

To further assist farmers, a separate knowledge center can be included in the app that offers agricultural advice, government scheme information, weather updates, and tutorials. This knowledge section will enable farmers with necessary information and enhance their digital literacy.

REFERENCES

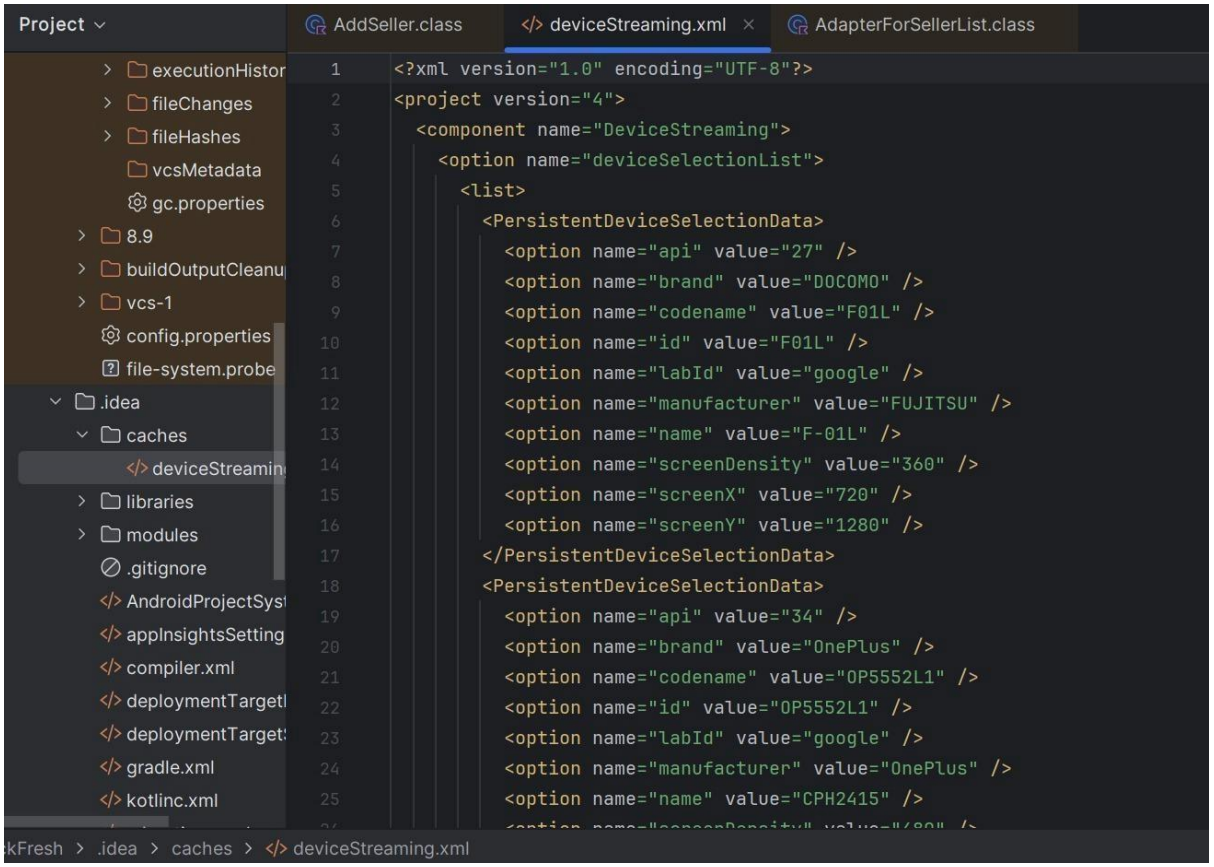
1. Customer demanding products in online shopping — A novel framework; 16-18 February 2017.
2. Hafizur Rahman; Eshan Barua; Samanta Afrin; Ashikur Rahman; Mohammad Monirujjaman Bid & Buy An Effective Online Based Platform for Client and Vendor; 08-10 April 2021.
3. Vincentius Riandaru Prasetyo; Searching Cheapest Product on Three Different E-Commerce Using K-Means Algorithm; 30-31 August 2018.
4. Thirumalai, S., & Sinha, K. K. (2005). Customer satisfaction with order fulfillment in retail supply chains: implications of product type in electronic B2C transactions. Journal of Operations Management, 23(3), 291-303.
5. Lee, J., Park, D. H., & Han, I. (2008). The effect of negative online consumer reviews on product attitude: An information processing view. Electronic commerce research and applications, 7(3), 341-352.
6. Kauffman, R. J., & Wang, B. (2002). Bid together, buy together: On the efficacy of group-buying business models in Internet-based selling. Handbook of Electronic Commerce in Business and Society, 99-137.
7. Maes, P., Guttman, R. H., & Moukas, A. G. (1999). Agents that buy and sell. Communications of the ACM, 42(3), 81-ff.
8. Budish, E. B., & Takeyama, L. N. (2001). Buy prices in online auctions: irrationality on the internet?. Economics letters, 72(3), 325-333.
9. Angst, C. M., Agarwal, R., & Kuruzovich, J. (2008). Bid or buy? Individual shopping traits as predictors of strategic exit in on-line auctions. International Journal of Electronic Commerce, 13(1), 59-84.
10. Shehryar, O. (2008). The effect of buyer's gender, risk-proneness, and time remaining in an internet auction on the decision to bid or buy-itnow. Journal of Product & Brand Management, 17(5), 356-365.
11. Ma, K. Du, F. Zheng, L. Zhang, and Z. Sun, "A segmentation method for processing greenhouse vegetable foliar disease symptom images," Information Processing in Agriculture, vol. 6, no. 2, pp. 216–223, 2019.
12. S. Khan and M. Narvekar, "Novel fusion of color balancing and superpixel based approach for detection of tomato plant diseases in natural complex environment," Journal of King

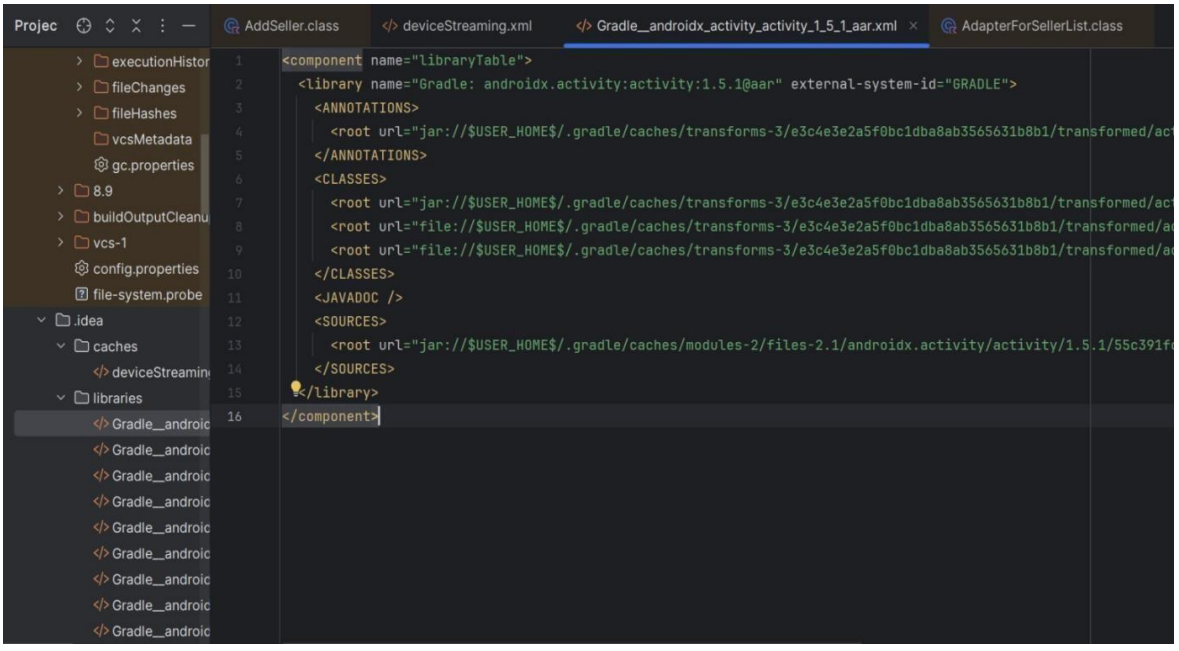
Saud University - Computer and Information Sciences, vol. 34, no. 6, Part B, pp. 3506– 3516, 2022.

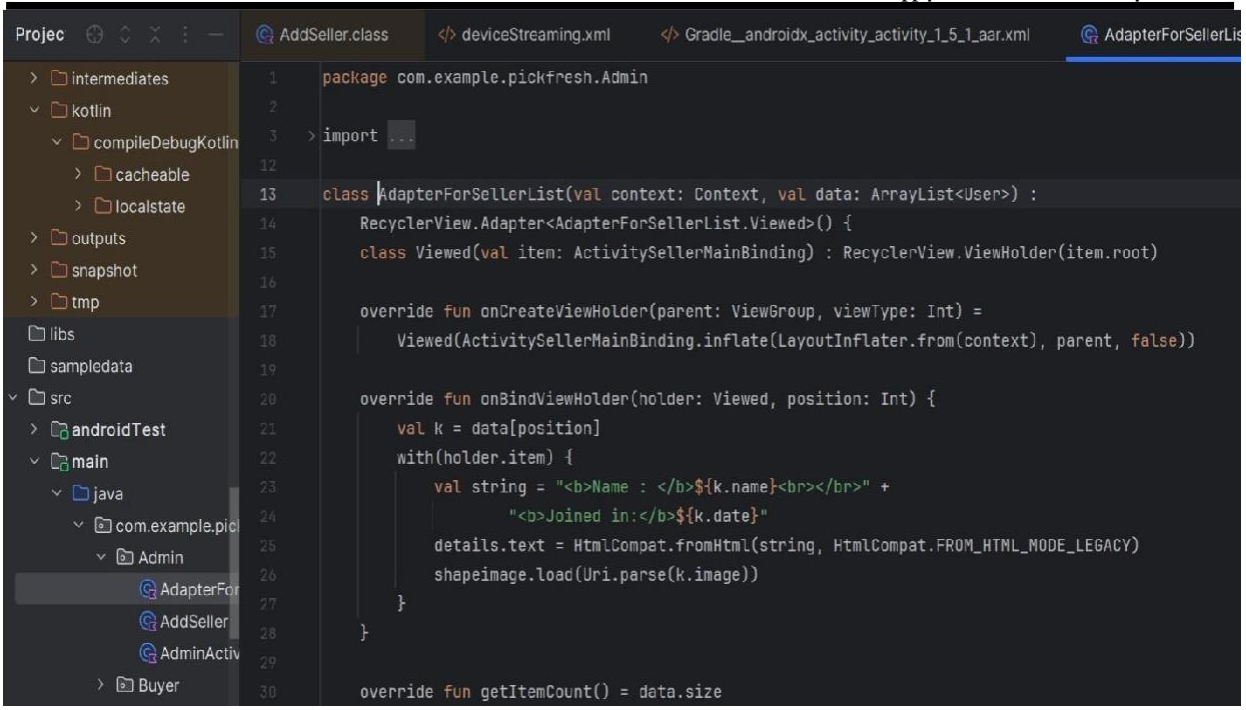
13. N. K. Trivedi, V. Gautam, A. Anand, H. M. Aljahdali, S. G. Villar, D. Anand, N. Goyal, and S. Kadry, "Early detection and classification of tomato leaf disease using high-performance deep neural network," Sensors, vol. 21, no. 23, pp. 1424–8220, 2021.
14. R. Sagar Vetal, "Tomato plant disease detection using image processing," International Journal of Advanced Research in Computer and Communication Engineering, vol. 6, no. 6, pp. 2278–1021, 2017.
15. S. K. Yusuke Kawasaki, Hiroyuki Uga and H. Iyatomi, "Basic study of automated diagnosis of viral plant diseases using convolutional neural networks," Advances in Visual Computing, pp. 638–645, 2015
16. Al-Hiary, Heba, Bani-Ahmad, Sulieman, Ryalat, Mohammad, Braik, Malik, Alrahamneh, and Zainab, "Fast and accurate detection and classification of plant diseases," International Journal of Computer Applications, vol. 517, no. 10.5120/2183-2754, pp. 46–129, 2020.
17. S. Mohanty, D. Hughes, and M. Salathe, "Using deep learning for image-based plant disease detection," Frontiers in Plant Science, vol. 7, no. 10.3389/fpls.2016.01419, pp. 46–129, 2016.
18. S. e. A. Raza, G. Prince, J. Clarkson, and N. Rajpoot, "Automatic detection of diseased tomato plants using thermal and stereo visible light images," Frontiers in Plant Science, vol. 10, no. e0123262. 10.1371/journal.pone.0123262, pp. 46–129, 2015.
19. G. Owomugisha, J. Quinn, E. Mwebaze, and J. Lwasa, "Automated vision-based diagnosis of banana bacterial wilt disease and black sigatoka disease," Frontiers in Plant Science, vol. 10, no. e0123262. 10.1371/journal.pone.0123262, pp. 46–129, 2019.
20. P. A, S. Chakraborty, A. Kumar, and O. Pooniwala, "Intelligent crop recommendation system using machine learning," Frontiers in Plant Science, vol. 10, no. 843-848. 10.1109/ICCMC51019.2021.9418375, pp. 46–129, 2021.

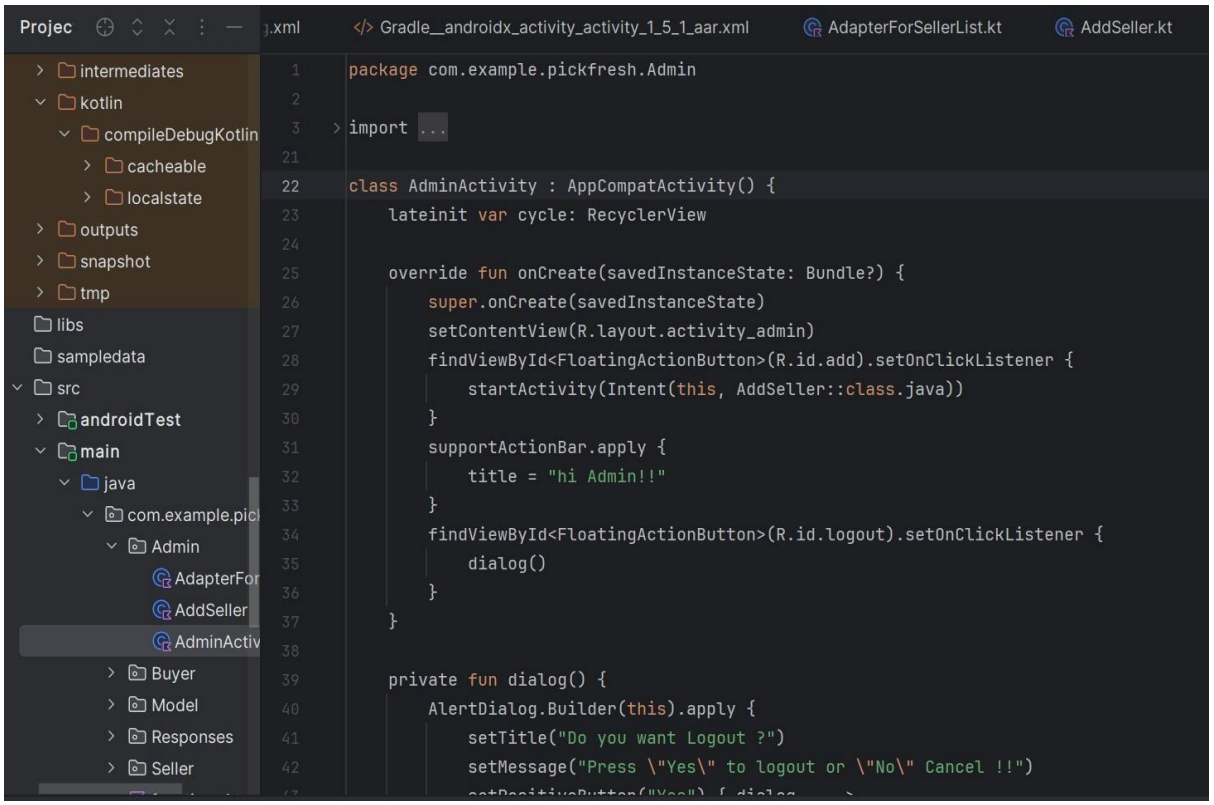
APPENDIX-A

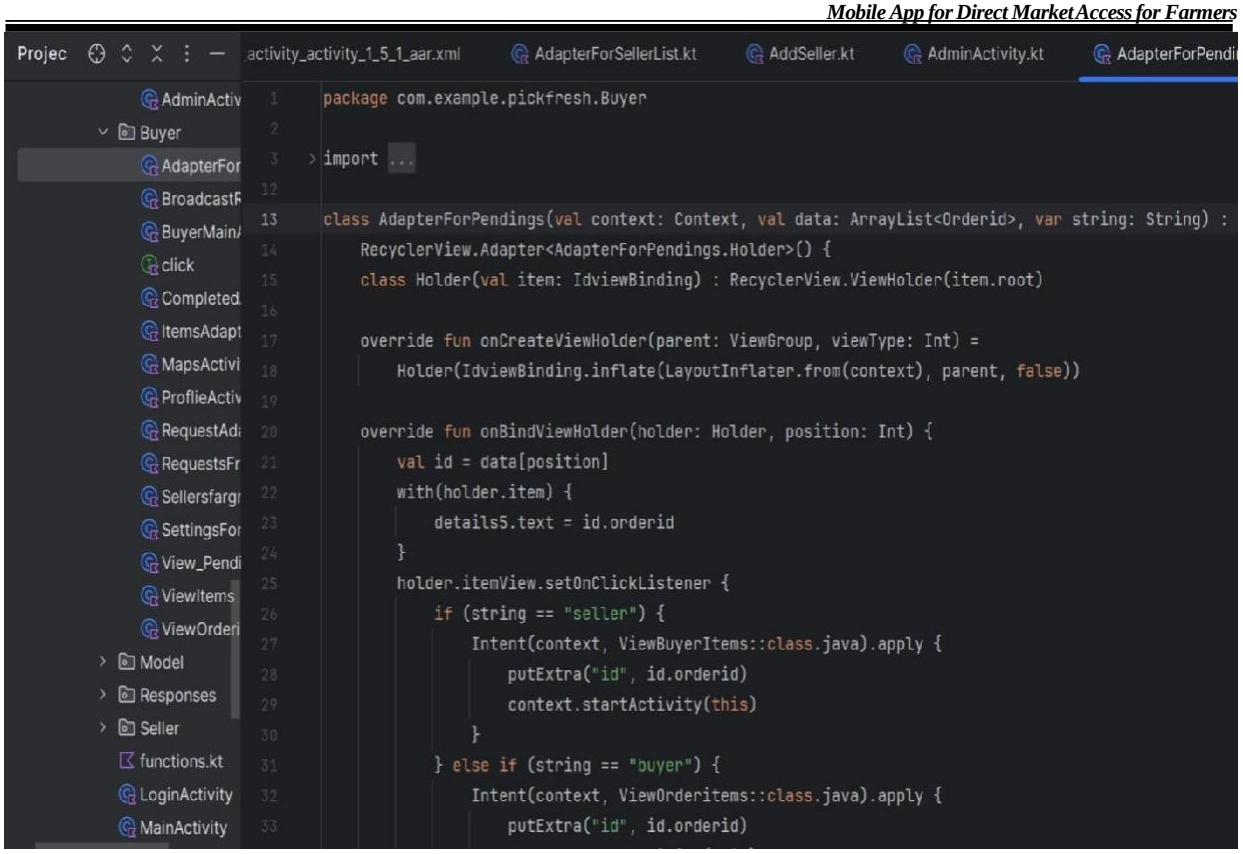
PSUEDOCODE

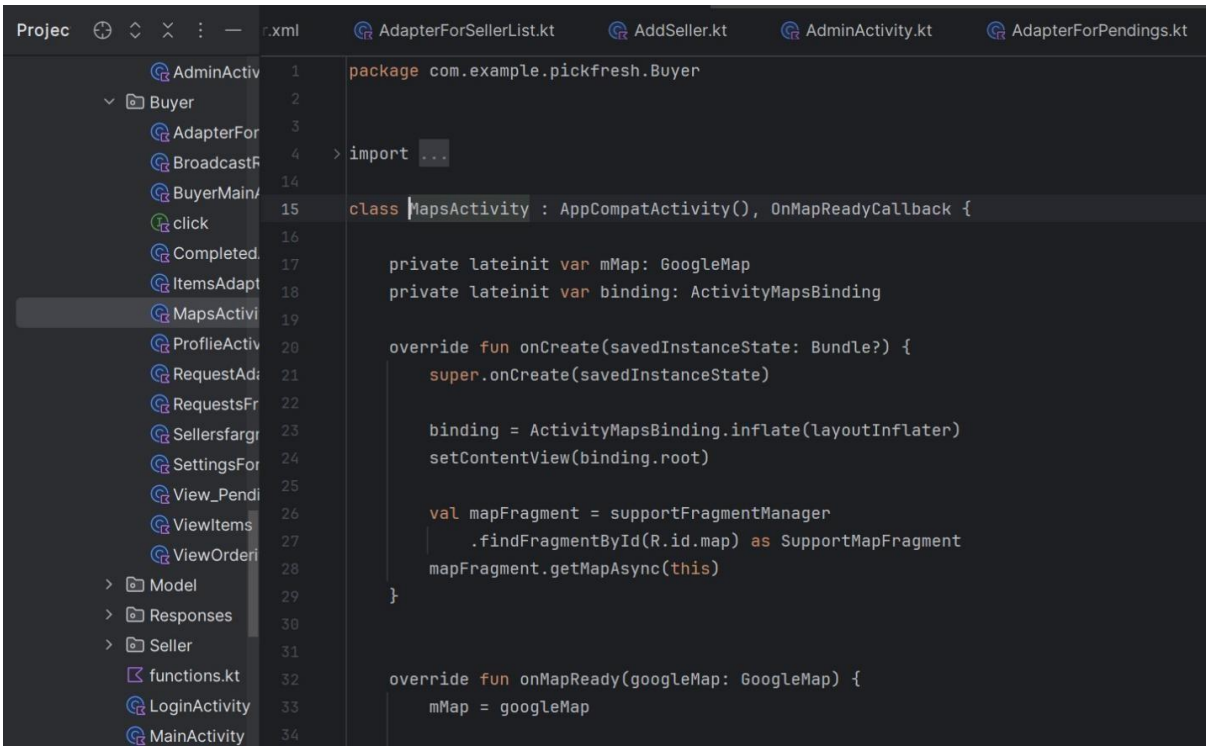


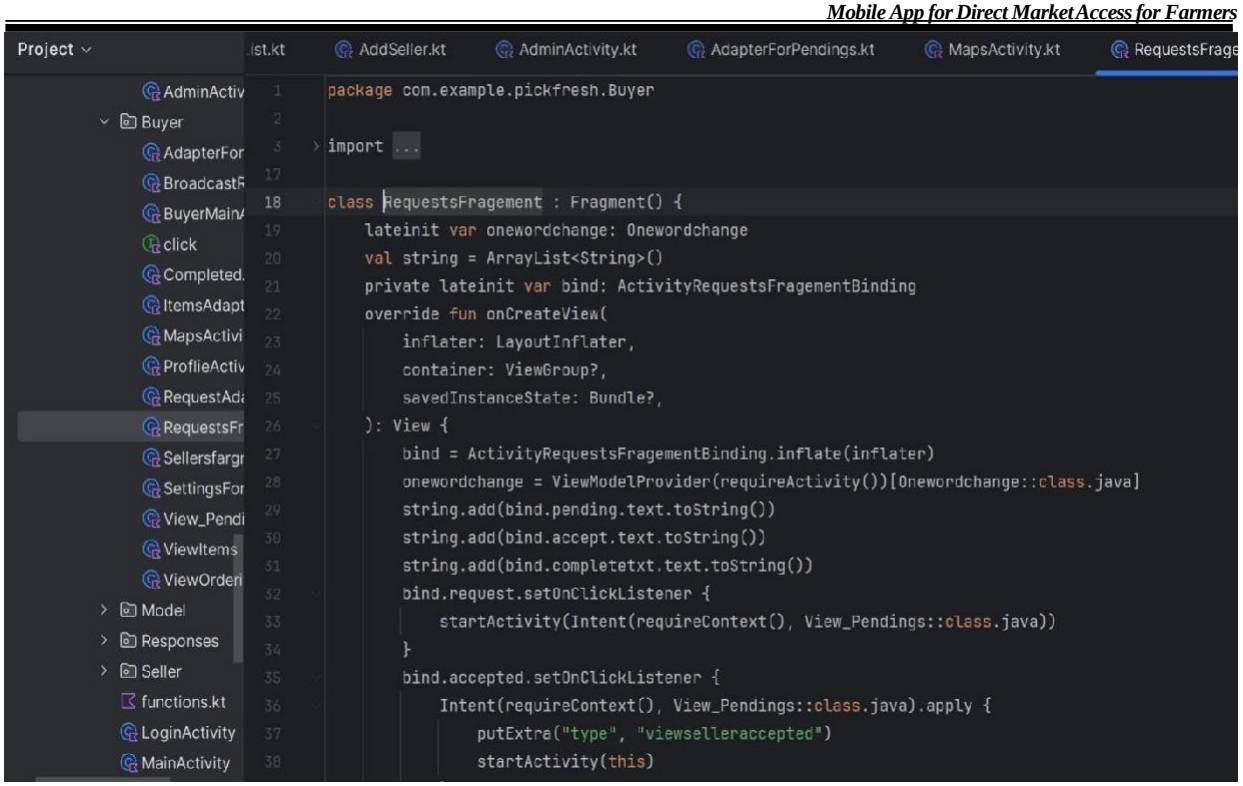


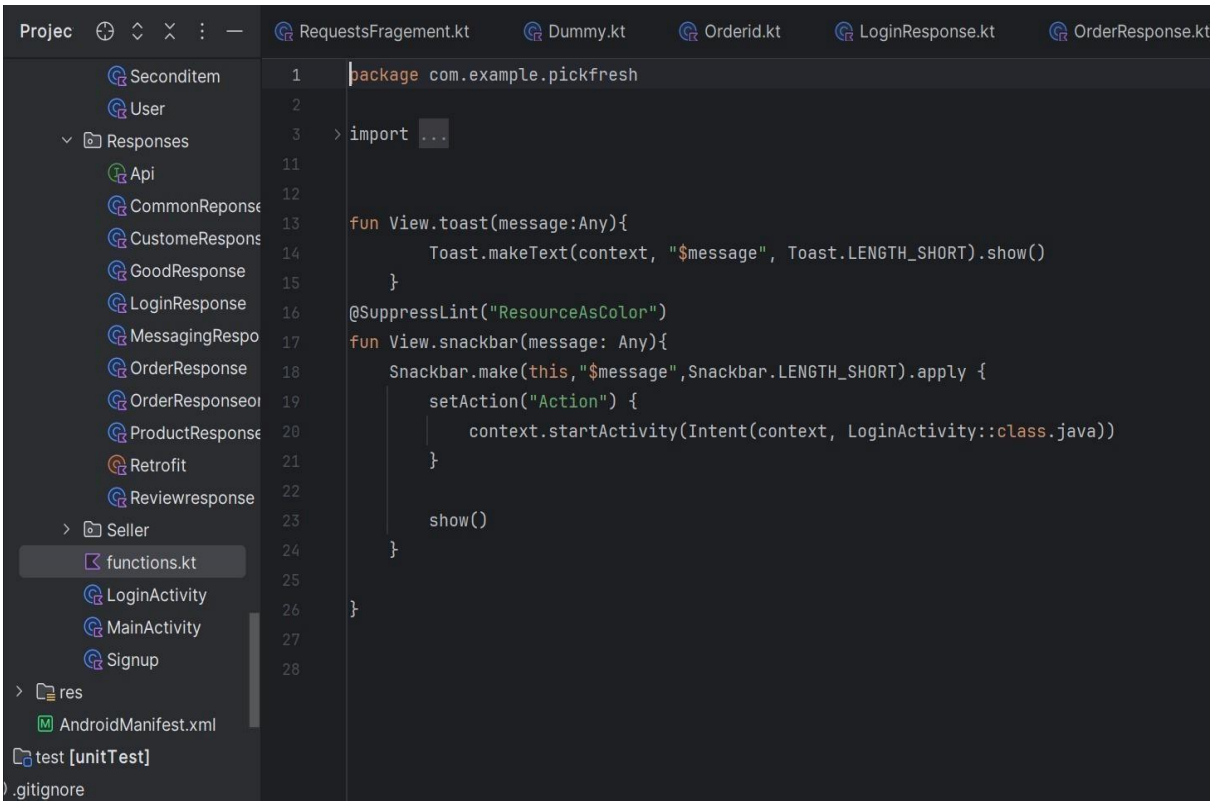

Projec
AddSeller.class
deviceStreaming.xml
Gradle__androidx_activity_activity_1_5_1_aar.xml
AdapterForSellerLis
intermediates
kotlin
compileDebugKotlin
cacheable
localstate
outputs
snapshot
tmp
libs
sampledata
src
androidTest
main
java
com.example.pic
Admin
AdapterFor
AddSeller
AdminActiv
Buyer
package com.example.pickfresh.Admin
import ...
class AdapterForSellerList(val context: Context, val data: ArrayList<User>) :
RecyclerView.Adapter<AdapterForSellerList.Viewed>() {
class Viewed(val item: ActivitySellerMainBinding) : RecyclerView.ViewHolder(item.root)
override fun onCreateViewHolder(parent: ViewGroup, viewType: Int) =
Viewed(ActivitySellerMainBinding.inflate(LayoutInflater.from(context), parent, false))
override fun onBindViewHolder(holder: Viewed, position: Int) {
val k = data[position]
with(holder.item) {
val string = "Name : ${k.name}
</br>" +
"Joined in:${k.date}"
details.text = HtmlCompat.fromHtml(string, HtmlCompat.FROM_HTML_MODE_LEGACY)
shapeimage.load(Uri.parse(k.image))
}
}
override fun getItemCount() = data.size

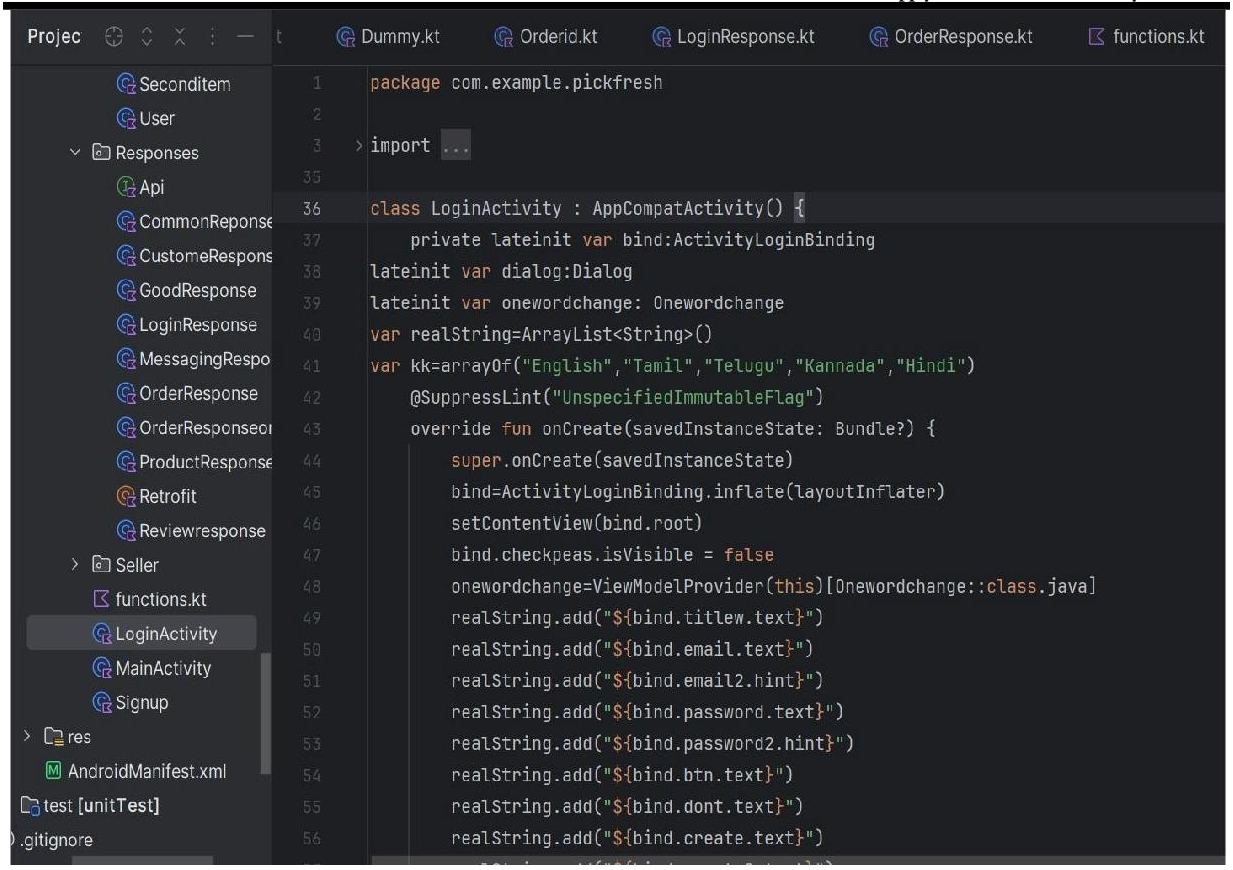

Projec
.xml
Gradle__androidx_activity_activity_1_5_1_aar.xml
AdapterForSellerList.kt
AddSeller.kt
intermediates
kotlin
compileDebugKotlin
cacheable
localstate
outputs
snapshot
tmp
libs
sampledata
src
androidTest
main
java
com.example.pic
Admin
AdapterFor
AddSeller
AdminActiv
Buyer
Model
Responses
Seller
package com.example.pickfresh.Admin
import ...
class AdminActivity : AppCompatActivity() {
lateinit var cycle: RecyclerView
override fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)
setContentView(R.layout.activity_admin)
findViewById<FloatingActionButton>(R.id.add).setOnClickListener {
startActivity(Intent(this, AddSeller::class.java))
}
supportActionBar.apply {
title = "hi Admin!!"
}
findViewById<FloatingActionButton>(R.id.logout).setOnClickListener {
dialog()
}
}
private fun dialog() {
AlertDialog.Builder(this).apply {
setTitle("Do you want Logout ?")
setMessage("Press \"Yes\" to logout or \"No\" Cancel !!")


Projec
activity_activity_1_5_1_aar.xml
AdapterForSellerList.kt
AddSeller.kt
AdminActivity.kt
AdapterForPendi
AdminActiv
Buyer
AdapterFor
BroadcastR
BuyerMainA
click
Completed
ItemsAdapt
MapsActivi
ProflieActiv
RequestAda
RequestsFr
Sellersfargr
SettingsFor
View_Pendi
ViewItems
ViewOrderi
Model
Responses
Seller
functions.kt
LoginActivity
MainActivity
package com.example.pickfresh.Buyer
import ...
class AdapterForPendings(val context: Context, val data: ArrayList<Orderid>, var string: String) :
RecyclerView.Adapter<AdapterForPendings.Holder>() {
class Holder(val item: IdviewBinding) : RecyclerView.ViewHolder(item.root)
override fun onCreateViewHolder(parent: ViewGroup, viewType: Int) =
Holder(IdviewBinding.inflate(LayoutInflater.from(context), parent, false))
override fun onBindViewHolder(holder: Holder, position: Int) {
val id = data[position]
with(holder.item) {
details5.text = id.orderid
}
holder.itemView.setOnClickListener {
if (string == "seller") {
Intent(context, ViewBuyerItems::class.java).apply {
putExtra("id", id.orderid)
context.startActivity(this)
}
} else if (string == "buyer") {
Intent(context, ViewOrderitems::class.java).apply {
putExtra("id", id.orderid)


Projec
.xml
AdapterForSellerList.kt
AddSeller.kt
AdminActivity.kt
AdapterForPendings.kt
AdminActiv
Buyer
AdapterFor
BroadcastR
BuyerMainA
click
Completed
ItemsAdapt
MapsActivi
ProflieActiv
RequestAda
RequestsFr
Sellersfargr
SettingsFor
View_Pendi
ViewItems
ViewOrderi
Model
Responses
Seller
functions.kt
LoginActivity
MainActivity
package com.example.pickfresh.Buyer
import ...
class MapsActivity : AppCompatActivity(), OnMapReadyCallback {
private lateinit var mMap: GoogleMap
private lateinit var binding: ActivityMapsBinding
override fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)
binding = ActivityMapsBinding.inflate(layoutInflater)
setContentView(binding.root)
val mapFragment = supportFragmentManager
.findFragmentById(R.id.map) as SupportMapFragment
mapFragment.getMapAsync(this)
}
override fun onMapReady(googleMap: GoogleMap) {
mMap = googleMap


Project
AddSeller.kt
AdminActivity.kt
AdapterForPendings.kt
MapsActivity.kt
RequestsFrage
AdminActiv
Buyer
AdapterFor
BroadcastR
BuyerMainA
click
Completed.
ItemsAdapt
MapsActivi
ProflieActiv
RequestAda
RequestsFr
Sellersfargr
SettingsFor
View_Pendi
ViewItems
ViewOrderi
Model
Responses
Seller
functions.kt
LoginActivity
MainActivity
package com.example.pickfresh.Buyer
import ...
class RequestsFragement : Fragment() {
lateinit var onewordchange: Onewordchange
val string = ArrayList<String>()
private lateinit var bind: ActivityRequestsFragementBinding
override fun onCreateView(
inflater: LayoutInflater,
container: ViewGroup?,
savedInstanceState: Bundle?,
): View {
bind = ActivityRequestsFragementBinding.inflate(inflater)
onewordchange = ViewModelProvider(requireActivity())[Onewordchange::class.java]
string.add(bind.pending.text.toString())
string.add(bind.accept.text.toString())
string.add(bind.completetxt.text.toString())
bind.request.setOnClickListener {
startActivity(Intent(requireContext(), View_Pendings::class.java))
}
bind.accepted.setOnClickListener {
Intent(requireContext(), View_Pendings::class.java).apply {
putExtra("type", "viewselleraccepted")
startActivity(this)


Projec
RequestsFragement.kt
Dummy.kt
Orderid.kt
LoginResponse.kt
OrderResponse.kt
Seconditem
User
Responses
Api
CommonReponse
CustomeRespons
GoodResponse
LoginResponse
MessagingRespo
OrderResponse
OrderResponseor
ProductResponse
Retrofit
Reviewresponse
Seller
functions.kt
LoginActivity
MainActivity
Signup
res
AndroidManifest.xml
test [unitTest]
.gitignore
package com.example.pickfresh
import ...
fun View.toast(message:Any){
Toast.makeText(context, "$message", Toast.LENGTH_SHORT).show()
}
@SuppressLint("ResourceAsColor")
fun View.snackbar(message: Any){
Snackbar.make(this,"$message",Snackbar.LENGTH_SHORT).apply {
setAction("Action") {
context.startActivity(Intent(context, LoginActivity::class.java))
}
show()
}
}


Projec
Dummy.kt
Orderid.kt
LoginResponse.kt
OrderResponse.kt
functions.kt
Seconditem
User
Responses
Api
CommonReponse
CustomeRespons
GoodResponse
LoginResponse
MessagingRespo
OrderResponse
OrderResponseor
ProductResponse
Retrofit
Reviewresponse
Seller
functions.kt
LoginActivity
MainActivity
Signup
res
AndroidManifest.xml
test [unitTest]
.gitignore
1 package com.example.pickfresh
3 import ...
36 class LoginActivity : AppCompatActivity() {
37 private lateinit var bind:ActivityLoginBinding
38 lateinit var dialog:Dialog
39 lateinit var onewordchange: Onewordchange
40 var realString=ArrayList<String>()
41 var kk=arrayOf("English","Tamil","Telugu","Kannada","Hindi")
42 @SuppressLint("UnspecifiedImmutableFlag")
43 override fun onCreate(savedInstanceState: Bundle?) {
44 super.onCreate(savedInstanceState)
45 bind=ActivityLoginBinding.inflate(layoutInflater)
46 setContentView(bind.root)
47 bind.checkpeas.isVisible = false
48 onewordchange=ViewModelProvider(this)[Onewordchange::class.java]
49 realString.add("${bind.titlew.text}")
50 realString.add("${bind.email.text}")
51 realString.add("${bind.email2.hint}")
52 realString.add("${bind.password.text}")
53 realString.add("${bind.password2.hint}")
54 realString.add("${bind.btn.text}")
55 realString.add("${bind.dont.text}")
56 realString.add("${bind.create.text}")

APPENDIX-B

OUTPUT SCREENSHOTS

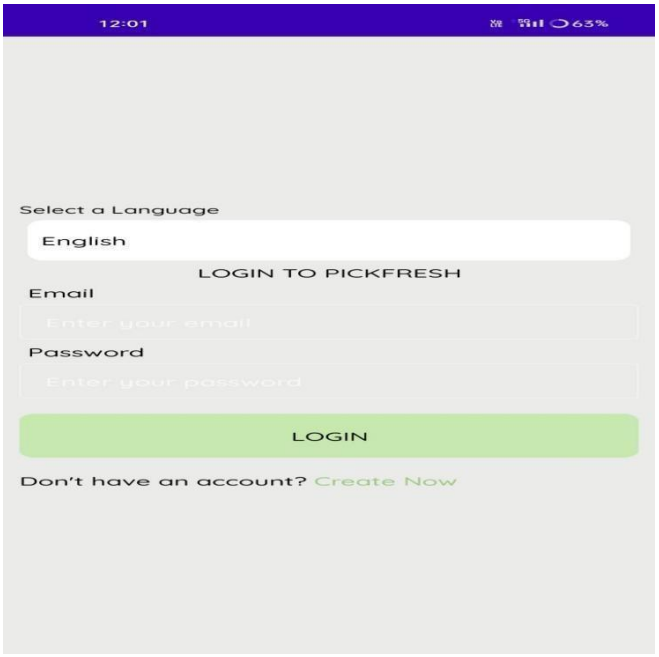


Fig No. A-B.1: Home Page

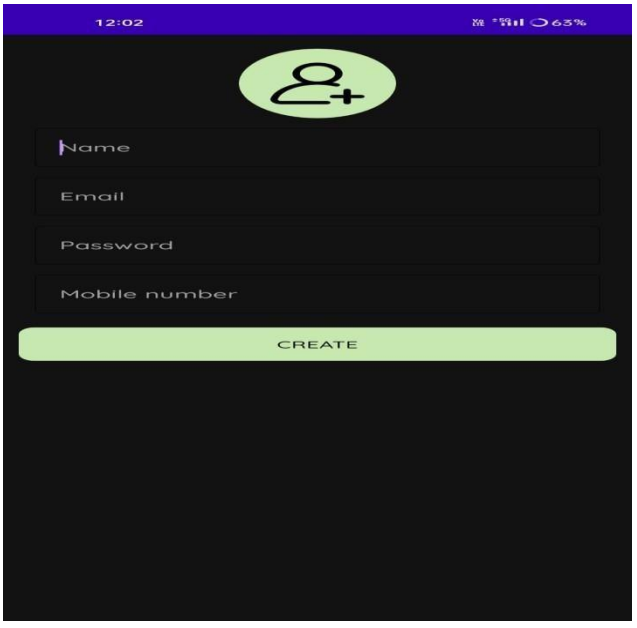


Fig No. A-B.2: Home Page

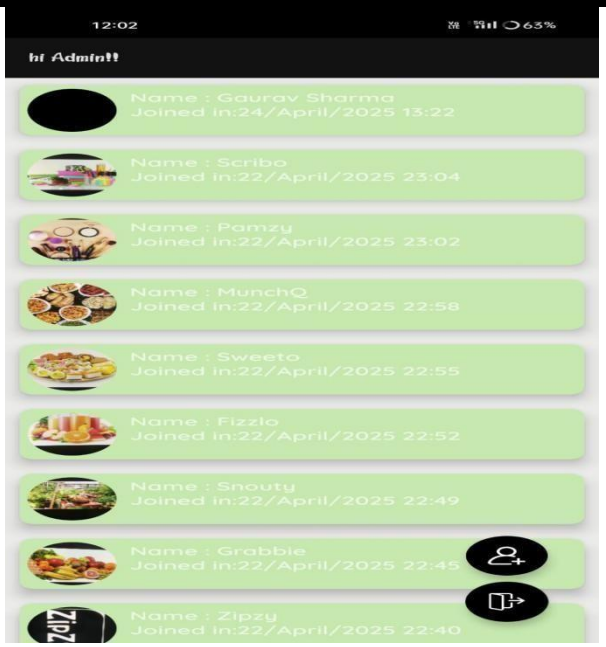


Fig No. A-B.3: SELLER ACCOUNTS

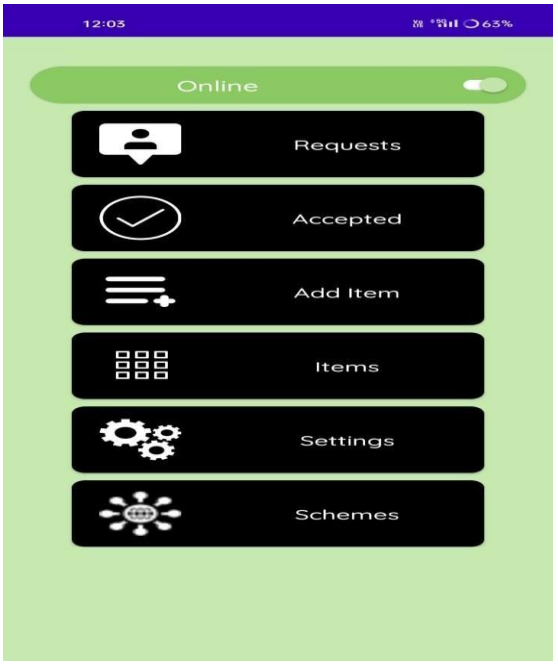


Fig No. A-B.4: SELLER SIDE FEATURES

Fig No. A-B.5: SCHEMES

Fig No. A-B-6: LOCATIONS OF SELLERS

APPENDIX-C
ENCLOSURES

SDG 1: No Poverty

Problem: Small-scale sellers and local farmers are inadequately integrated into the market, limiting their income and economic prospects.

Pick Fresh Solution: The app allows buyers and sellers within the same locality to transact directly, eliminating the need for middlemen.

Impact: Enhances financial income opportunities for sellers, fosters financial inclusion, and reduces poverty in rural and semi-urban areas.

SDG 8: Decent Work and Economic Growth

Challenge: Small vendors and farmers face every barrier to entering digital marketplaces including lack of infrastructure and exposure.

Pick Fresh Approach: The app provides a mobile-first approach with multilingual offerings and a user-friendly interface that creates and enables the participation of local entrepreneurs and sellers in e-commerce.

Impact: Increases local employment, celebrates small business, and supports global economic inclusiveness.

SDG 9: Industry, Innovation and Infrastructure

Issue: Informal sellers are disconnected from modern digital commerce and have no access to formal selling tools.

Pick Fresh Approach: The app uses location technology, real-time product management, and scheme integration to provide a smart, scalable, localized trade infrastructure.

Outcome: Supports innovation in rural commerce, digitizes traditional markets and creates sustainable infrastructure for local industries.

SDG 10: Reduced Inequalities

Problem: Digital marketplaces often create access issues for marginalized communities when there is language or digital illiteracy problems.

Pick Fresh Approach: The app includes multilingual support (Tamil, Telugu, Kannada, Hindi, English) and is designed to be usable by people with low literacy levels.

Results: Facilitate equitable participation, reduce digital inequality, and empower underserved communities in the digital economy.

SDG 11: Sustainable Cities and Communities

Issue: Urban food supply chains often overlook local vendors and are dominated by large retailers.

Pick Fresh Approach: The app enables city residents to find and support nearby local sellers, reducing food miles and strengthening local economies.

Outcome: Fosters sustainable urban-rural linkages and supports community-based commerce networks.

Plagiarism Report

Ramamurthy Ketha cbc02Report

ORIGINALITY REPORT

14%	13%	2%	13%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to North Ft. Myers High School Student Paper	3%
2	Submitted to Presidency University Student Paper	3%
3	www.jetir.org Internet Source	2%
4	www.caedes.net Internet Source	2%
5	pdfcoffee.com Internet Source	1%
6	Submitted to Manipal University Student Paper	1%
7	Submitted to M S Ramaiah University of Applied Sciences Student Paper	<1%
8	Submitted to Symbiosis International University Student Paper	<1%
9	Submitted to Jawaharlal Nehru Technological University Anantapur Student Paper	<1%
10	Submitted to Southern New Hampshire University - Continuing Education Student Paper	<1%
11	Submitted to Jawaharlal Nehru Technological University Kakinada Student Paper	<1%
12	www.slideshare.net Internet Source	<1%
13	Submitted to srmap Student Paper	<1%