A
Project Report
On
"Quick Eat"

Prepared by
Dharmjeet Vala (D22DCS159)
Dhrumil Bhatiya (D22DCS160)
Vandan Patel (D22DCS161)

Under the guidance of
Prof. Gaurang Patel
Assistant Professor

A Report Submitted to
Charotar University of Science and Technology
for Partial Fulfillment of the Requirements for the
6th Semester Software Group Project-IV(CS-357)

Submitted at

Accredited with Grade A+ by NAAC

Computer Science & Engineering
Devang Patel Institute of Advanced Technology and Research
At: Changa, Dist: Anand – 388421
2024

CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CERTIFICATE

This is to certify that the report entitled "**Quick Eat**" is a bonafied work carried out by **Mr. Dharmjeet Vala (D22DCS159)** under the guidance and supervision of **Prof. Gaurang Patel** for the subject **CS-357** - **Software Group Project-IV**(CSE) of 6th Semester of Bachelor of Technology in **DEPSTAR** at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Prof. Gaurang Patel
Assistant Professor
Computer Science & Engineering
DEPSTAR, Changa, Gujarat.

Dr. Chirag Patel
Head of Department – CSE,
DEPSTAR
CHARUSAT, Changa, Gujarat.

Devang Patel Institute of Advance Technology and Research At: Changa, Ta. Petlad, Dist. Anand, PIN: 388 421. Gujarat

CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CERTIFICATE

This is to certify that the report entitled "**Quick Eat**" is a bonafied work carried out by **Mr. Dhrumil Bhatiya (D22DCS160)** under the guidance and supervision of **Prof. Gaurang Patel** for the subject **CS-357** - **Software Group Project-IV**(CSE) of 6th Semester of Bachelor of Technology in **DEPSTAR** at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Prof. Gaurang Patel
Assistant Professor
Computer Science & Engineering
DEPSTAR, Changa, Gujarat.

Dr. Chirag Patel
Head of Department – CSE,
DEPSTAR
CHARUSAT, Changa, Gujarat.

Devang Patel Institute of Advance Technology and Research At: Changa, Ta. Petlad, Dist. Anand, PIN: 388 421. Gujarat

CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CERTIFICATE

This is to certify that the report entitled "**Quick Eat**" is a bonafied work carried out by **Mr. Vandan Patel (D22DCS161)** under the guidance and supervision of **Prof. Gaurang Patel** for the subject **CS-357** - **Software Group Project-IV**(CSE) of 6th Semester of Bachelor of Technology in **DEPSTAR** at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Prof. Gaurang Patel
Assistant Professor
Computer Science & Engineering
DEPSTAR, Changa, Gujarat.

Dr. Chirag Patel
Head of Department – CSE,
DEPSTAR
CHARUSAT, Changa, Gujarat.

Devang Patel Institute of Advance Technology and Research At: Changa, Ta. Petlad, Dist. Anand, PIN: 388 421. Gujarat

DECLARATION BY THE CANDIDATES

We hereby declare that the project report entitled "**Quick Eat"** submitted by me to Devang Patel Institute of Advance Technology and Research, Changa in partial fulfilment of the requirement for the award of the degree of **B. Tech.** in Computer Engineering, from Devang Patel Institute of Advance Technology and Research, DEPSTAR/FTE, is a record of bonafide CS-357 Software Project GROUP (project work) carried out by us under the guidance of **Prof. Gaurang Patel** We further declare that the work carried out and documented in this project report has not been submitted anywhere else either in part or in full and it is the original work, for the award of any other degree or diploma in this institute or any other institute or university.

Dharmjeet Vala (D22DCS159)   Dhrumil Bhatiya (D22DCS160)

Signature of student   **Signature of student**

Vandan Patel (D22DCS161)

Signature of Student

ACKNOWLEDGEMENT

We, the developer of a console-based game **"Quick Eat"**, with immense pleasure and commitment would like to present the project assignment. The development of this project has given us wide opportunity to think, implement and interact with various aspects of management skills as well as the new emerging technologies.

Every work that one completes successfully stands on the constant encouragement, good will and support of the people around. We hereby avail this opportunity to express us gratitude to number of people who extended their valuable time, full support and cooperation in developing the project.

We express deep sense of gratitude towards our Head of the CSE Department, Dr. Chirag Patel and project guide Prof. Gaurang Patel for the support during the whole session of study and development. It is because of them, that we were prompted to do hard work, adopting new technologies.

Thanks,

Dharmjeet Vala (D22DCS159)

Dhrumil Bhatiya (D22DCS160)

Vandan Patel (D22DCS161)

ABSTRACT

"Quick Eat" is an Android application developed with Kotlin in Android Studio, designed to simplify dietary management for users amidst their busy lives. It enables users to create personalized profiles outlining their dietary preferences, restrictions, and health objectives. Leveraging Kotlin's capabilities, the app generates tailored meal recommendations and offers a diverse recipe repository that aligns with users' dietary needs. It streamlines meal planning by integrating shopping lists based on selected recipes and tracks users' progress towards their health goals. Through intuitive navigation and user-friendly design, "Eat Sure" aims to empower users to make informed dietary choices and lead healthier lifestyles.

By harnessing Kotlin's versatility and Android Studio's development environment, "Eat Sure" offers a comprehensive solution for individuals seeking effective dietary management tools. The app's features include personalized meal recommendations, a curated recipe repository, automated shopping list generation, and progress tracking. With continuous refinement, "Eat Sure" strives to provide users with the resources they need to maintain balanced diets and achieve their health objectives, ultimately promoting a healthier lifestyle in today's fast-paced world.

Index

List of Figures

CHAPTER 1: INTRODUCTION

1.1 Project overview

The "Eat Sure" project is an innovative Android application developed using Kotlin in Android Studio, aimed at revolutionizing dietary management for users. With an increasingly fast-paced lifestyle, many individuals struggle to maintain healthy eating habits. "Eat Sure" addresses this challenge by providing a comprehensive solution that combines personalized dietary recommendations, a vast recipe repository, and streamlined meal planning features. Leveraging the power of Kotlin, the application offers users the ability to create detailed profiles outlining their dietary preferences, restrictions, and health goals. This information is then used to generate tailored meal recommendations that prioritize nutritional balance and align with individual preferences.

One of the key components of "Eat Sure" is its curated recipe repository, which offers users access to a diverse collection of recipes tailored to their dietary needs. Users can explore recipes based on various criteria such as cuisine, dietary restrictions, and preparation time, ensuring that they can find meals that suit their tastes and lifestyle. Additionally, the application integrates seamlessly with users' daily routines by automatically generating shopping lists based on selected recipes. This feature simplifies the meal planning process, allowing users to efficiently organize their grocery shopping and meal preparation activities.

Furthermore, "Eat Sure" incorporates progress tracking functionalities, enabling users to monitor their dietary adherence and track their progress towards their health goals. Through intuitive navigation and user-friendly design, the application ensures accessibility for users of all skill levels. With continuous refinement and updates, the "Eat Sure" project aims to empower users to make informed dietary choices and cultivate healthier lifestyles. By leveraging Kotlin's versatility and Android Studio's development environment, "Eat Sure" provides a comprehensive platform for individuals seeking to optimize their dietary management strategies in today's fast-paced world.

1.2 Objective

- **Restaurant Discovery**: Enable users to discover nearby restaurants based on their location, cuisine preferences, or specific dishes they're interested in trying.

- **Menu Exploration**: Provide users with access to restaurant menus, allowing them to browse through the available dishes, view descriptions, and see pricing information.

- **User Reviews and Ratings**: Incorporate a feature for users to read and write reviews and ratings for restaurants and individual dishes, helping others make informed dining decisions.

- **Reservation and Ordering**: Implement functionalities for users to make restaurant reservations or place orders for pickup or delivery directly through the app, streamlining the dining experience.

- **Special Offers and Promotions**: Partner with restaurants to showcase special offers, discounts, or promotions within the app, enticing users to explore new dining options and save money.

- **Customization Preferences**: Allow users to specify dietary preferences or restrictions (e.g., vegetarian, gluten-free) to filter restaurant options and find dishes that meet their needs.

- **Social Sharing**: Integrate social sharing features to enable users to share their dining experiences, favorite restaurants, or recommended dishes with friends and followers on social media platforms.

- **Restaurant Recommendations**: Utilize algorithms or user preferences to provide personalized restaurant recommendations based on past dining history or user preferences.

- **Location-Based Notifications**: Implement location-based notifications to alert users of nearby restaurants offering special deals or new menu items, enhancing user engagement.

- **Accessibility Features**: Ensure the app is accessible to users with disabilities by incorporating features such as voice commands, screen reader compatibility, and high contrast modes.

1.3 Scope

The scope of the application encompasses providing users with comprehensive information about various restaurants, including their names, locations, contact details, hours of operation, and types of cuisine offered. Users can view restaurant menus with descriptions and prices, search for restaurants based on location, cuisine type, price range, and user ratings, and filter menu items based on dietary preferences or restrictions. The app facilitates user interaction through review submissions, ratings, and photo sharing of dining experiences, as well as discussions and recommendations among users. Additionally, users can make reservations for dine-in or place orders for pickup or delivery directly through the app if supported by the restaurants. Notifications about special offers, discounts, or new menu items may also be provided to users.

1.4 Tools & Technology Used

➢ Android Studio

➢ Ad Mob

Device Compatibility:

• Any android device above 9 will be supported.

1.4 Tools & Technology Used

CHAPTER 2: PROJECT PLANNING

2.1 Project Development Approach and Justification
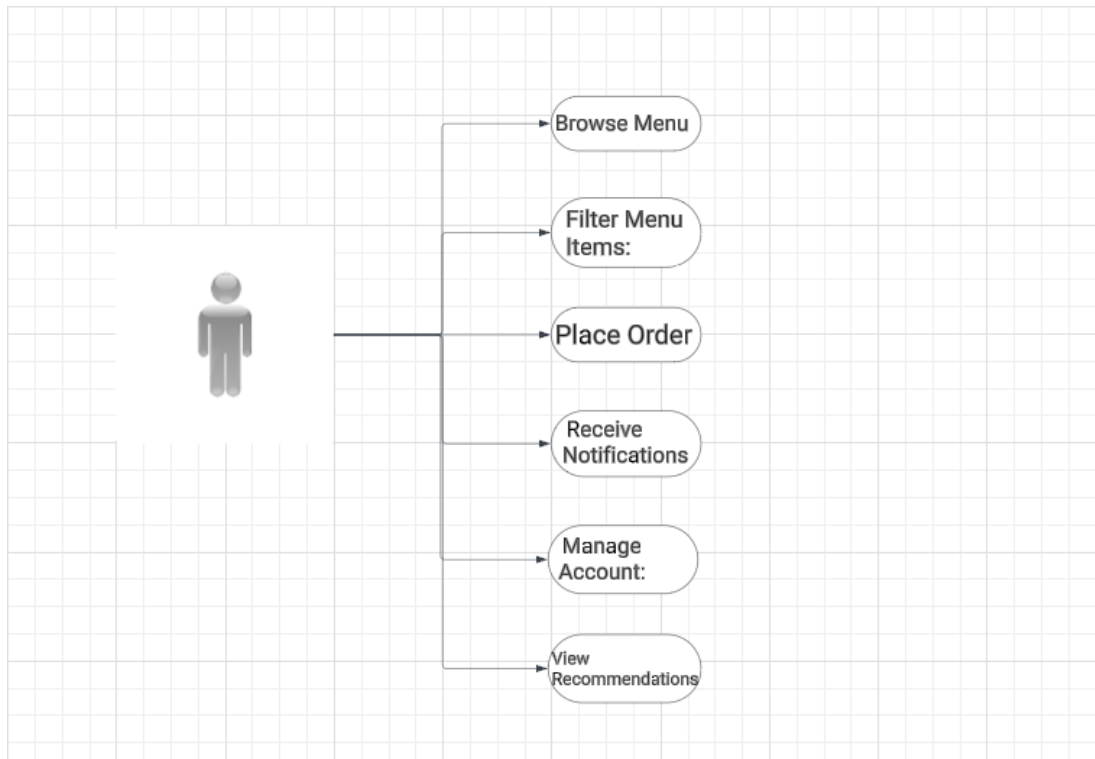
Figure 1 Use Case Diagram

(In this above use case diagram, we have shown the how the user will act with Quick Eat and by using the application the user will be able to browse the menu and also can filter the menu items after the user can place the order and can receive the notification of placed order)
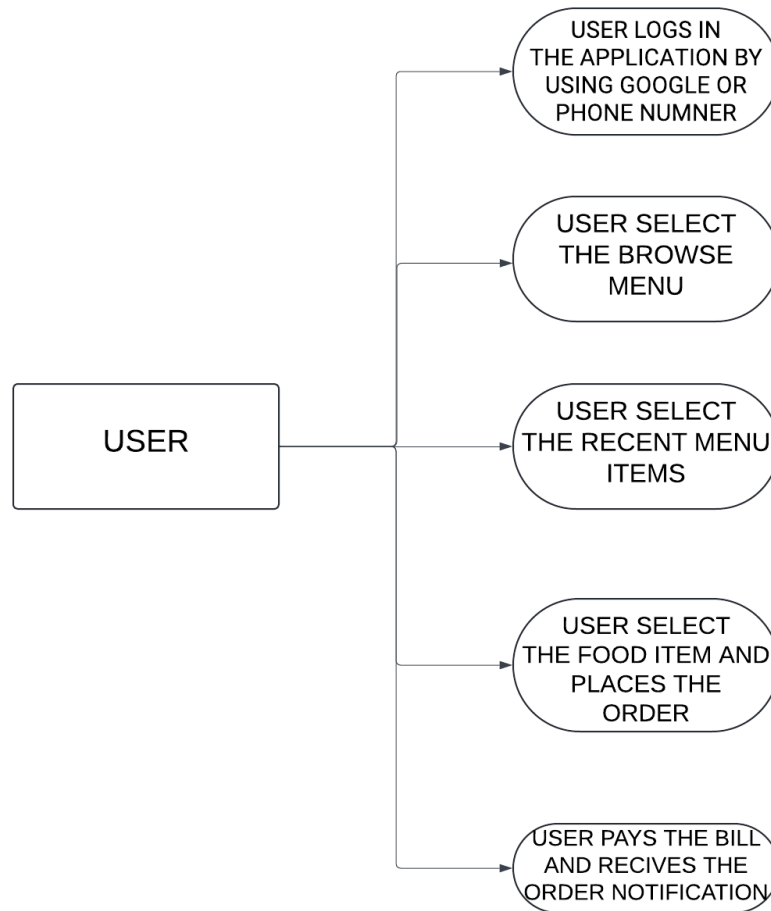
Figure 2 Data Flow Diagram

(The Data Flow Diagram for the Quick Eat visually represents the flow of data as users interact with the system's processes and external entities.)
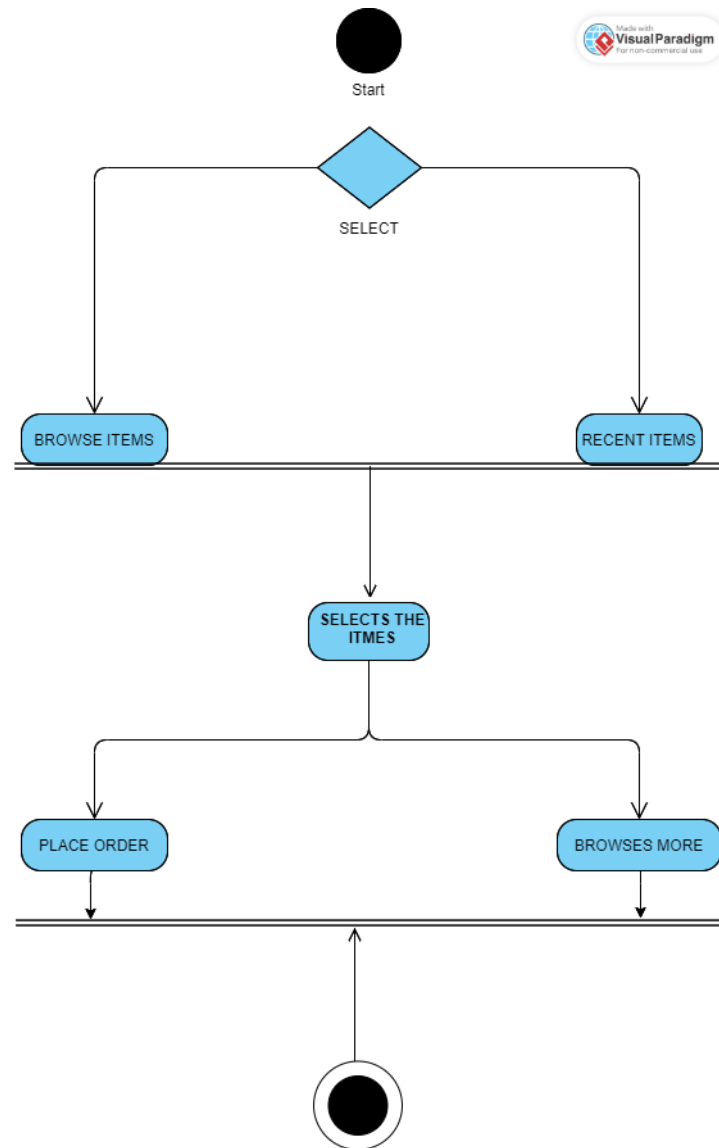
Figure 3 Activity Diagram

(The activity diagram provides a clear and concise representation of the sequential steps users take within the Quick Eat. It outlines the process of selecting, editing, ordering, and placing the order, guiding users through the various tasks and options available in the application.)
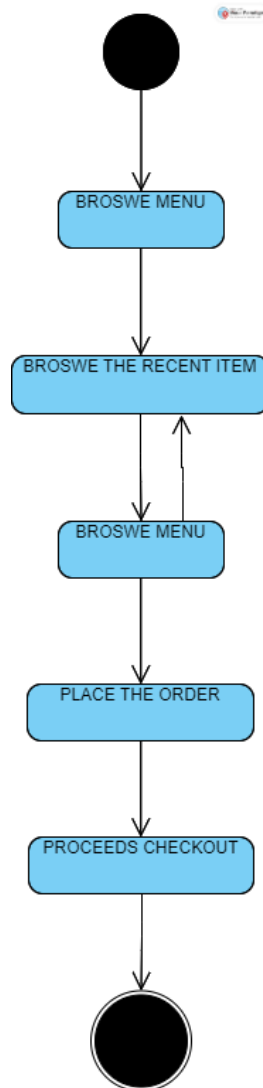
Figure 4 State Diagram

(The State Diagram for the Quick Eat visually represents the different states that the application can be in and the transitions between these states based on user interactions and internal events. It provides a comprehensive view of the application's behavior and helps developers understand how the application's state changes over time in response to various user actions.)

CHAPTER 3: SYSTEM REQUIRMENTS STUDY

3.1 USER CHARACTERSTICS

End Users:

- **Diverse Demographics:** The application is designed to cater to users of varying ages, backgrounds, and lifestyles, including individuals, families, and groups.
- **Food Enthusiasts**: Users who are passionate about food and dining experiences, seeking new restaurants, cuisines, and dishes to explore.
- **Tech-Savvy Individuals**: Users comfortable with using mobile applications and digital platforms for tasks such as restaurant discovery, menu browsing, and online ordering.
- **Busy Professionals**: Users with busy schedules who appreciate the convenience of quickly finding nearby restaurants, viewing menus, and making reservations or orders on the go.
- **Socializers:** Users who enjoy dining out with friends, family, or colleagues and may use the application to coordinate dining plans, share recommendations, and engage in discussions about restaurants and dishes.
- **Budget-Conscious Consumers**: Users who are mindful of their spending and seek value-for-money dining options, including special offers, discounts, and promotions.
- **Health-Conscious Individuals**: While the application doesn't focus on nutritional value or diet management, some users may still be interested in finding restaurants that offer healthy meal options or accommodate specific dietary restrictions.
- **Explorers and Travelers**: Users who are visiting new cities or areas and are interested in discovering local restaurants and cuisines.

3.2 HARDWARE AND SOFTWARE REQUIREMENTS

3.2.1 Hardware Specification

•   Minimum 4 GB RAM

•   Storage 64Gb internal Storage

3.2.2 Software Specification

•   OS: Android 9 or above

•   Snapdragon or MediaTek Processor up to 1.6Ghz

•   Latest with updates and patches of Android Version.

CHAPTER 4: SYSTEM ANALYSIS

4.1 STUDY OF PROPOSED SOLUTION

- Quick Eat's path also includes thorough documentation, the formation of customer support, and a complete marketing and launch plan. Following the launch, the application's performance is refined by monitoring use, assessing KPIs, and planning updates. This in-depth research guarantees that Quick Eat not only matches user expectations, but also thrives in the competitive Android photography and editing app market. It's a fluid process that adjusts to changing user preferences and market conditions, securing Quick Eat's position as a user-centric and popular mobile food ordering app.

- **Installing Android Studio 2024**

Download Android Studio:
- Visit the official Android Studio download page at https://developer.android.com/studio.
- Download the appropriate version for your operating system (Windows, macOS, or Linux).
- Run the installer and follow the on-screen instructions to install Android Studio on your computer.

Launch Android Studio
- After installation, launch Android Studio.

Android Studio Setup Wizard:
- The first time you launch Android Studio, it will open the Android Studio Setup Wizard.
- Click "Next" to begin the setup process.

Select Configuration:
- Choose the "Standard" installation option for a typical setup.
- Click "Next."

SDK Components Setup:
- Android Studio will download and install the Android SDK components.
- Ensure you have a working internet connection as this may take some time.

Accept License Agreements:
- Review and accept the terms of the license agreements for the SDK components.
- Click "Next."

Choose UI Theme:
- Select your preferred UI theme (Dark or Light).
- Click "Next."

Verify Settings:
- Review the settings and configurations.
- Click "Finish" to complete the setup.

Update and Restart:
- Android Studio may prompt you to update the IDE to the latest version.
- Follow the prompts to update and restart Android Studio.

Create a New Kotlin Project:
- After Android Studio is up to date, click on "Start a new Android Studio project."

Select Project Template:
- Choose a project template (e.g., "Empty Activity," "Basic Activity") and click "Next."

Configure Your Project:
- Provide a name for your project, package name, and location.
- Choose the language as "Kotlin."
- Select the minimum API level based on your target audience.
- Configure other project options as needed

- Click "Finish

Build and Run:
- Android Studio will create your new Kotlin project.
- Once the project is loaded, you can build and run it on an emulator or a physical Android device by clicking the "Run" button.

CHAPTER 5: SYSTEM DESIGN

User Side

Figure 6 Splash Screen of Application

(The above images show the splash screen of the application)

Figure 7 Login Page of Application

(The above image represents the login page of the application)
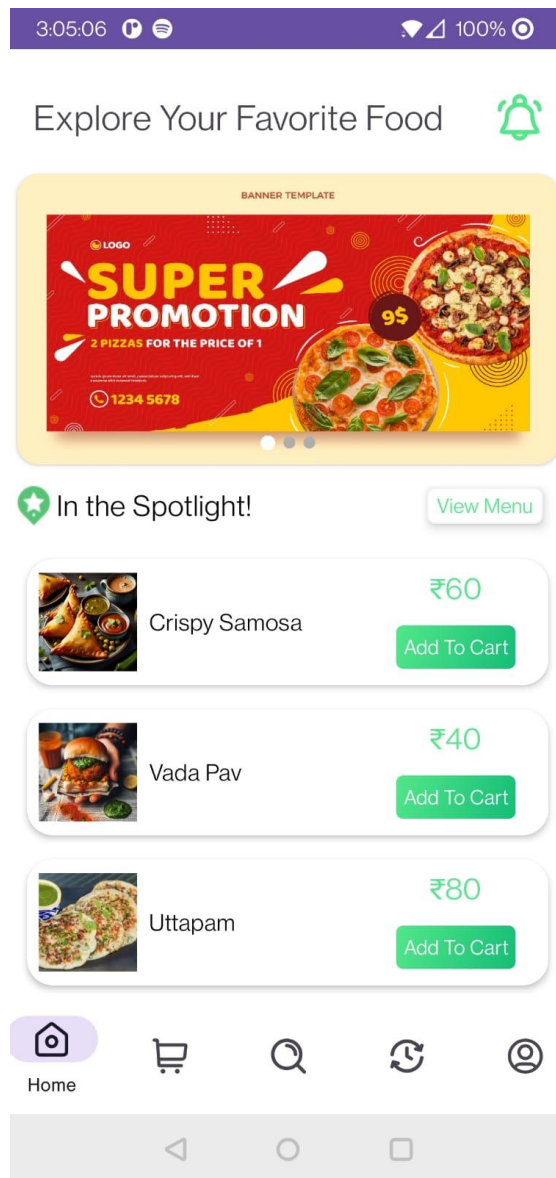
Figure 8 Browing Menu Screen

(The above images shows that the user is browsing the main menu)
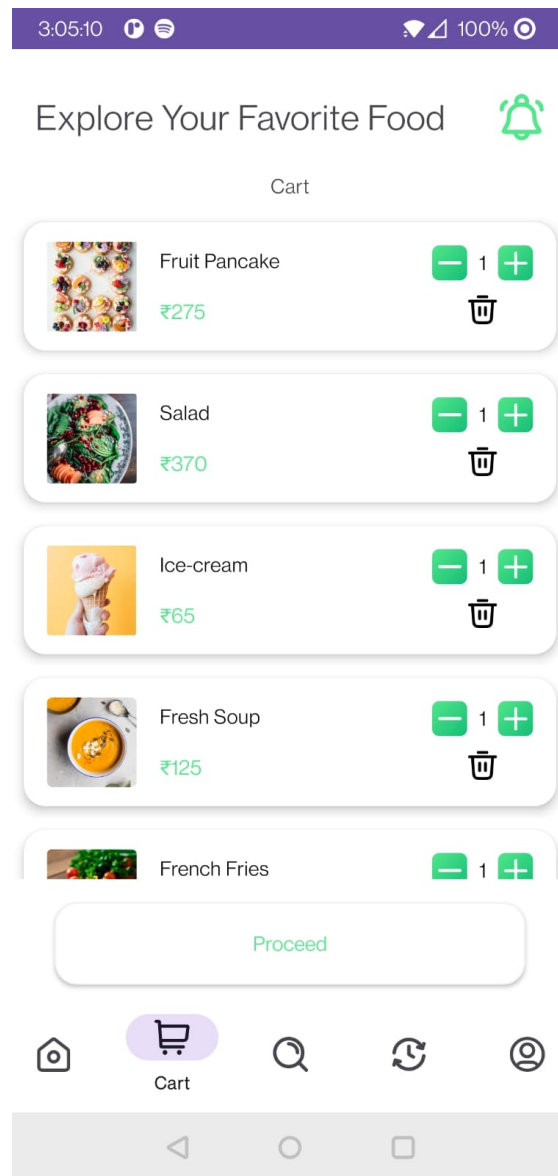
Figure 9 Recent Menu Screen

(Above image shows that user has clicked on the recent or favorite menu items)
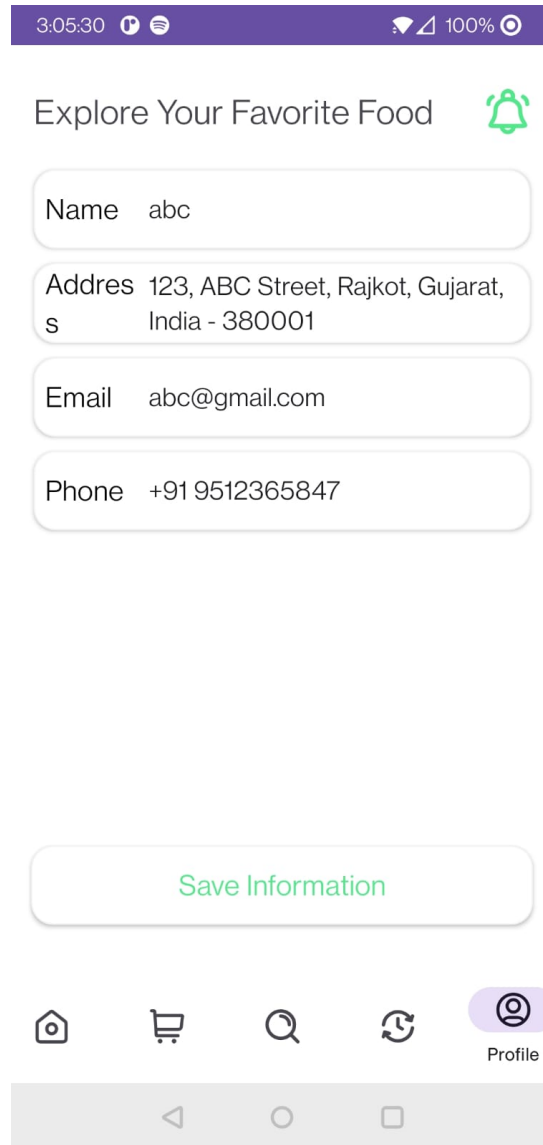
Figure 10 Checkout Page

(The above image shows that user has proceed to the checkout page)

Figure 11 User Profile

(The above images shows that user exploring the user profile)

Admin Side
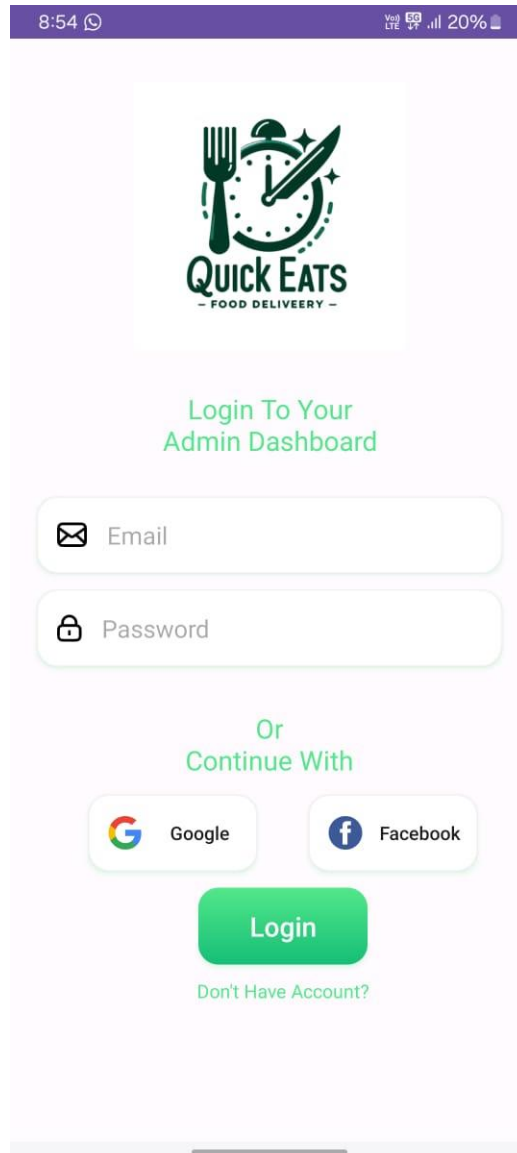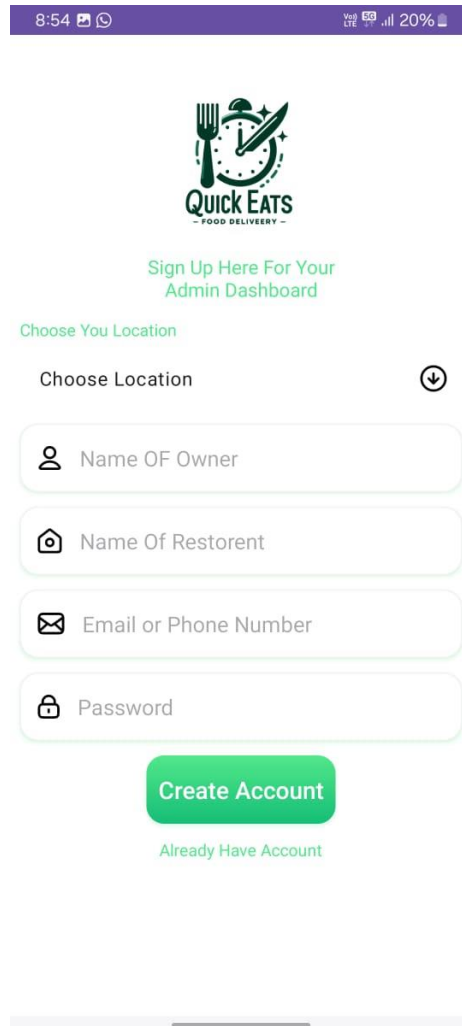
Figure 12 User Logins for the admin

(Above figure shows that the user is login for the admin side application or panel)

Figure 13 User Details

(Above Images shows that the user is filing the restaurant details)
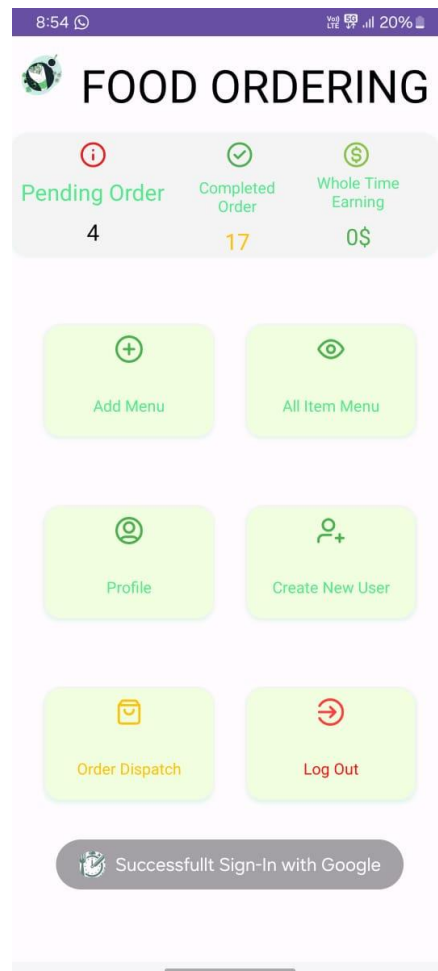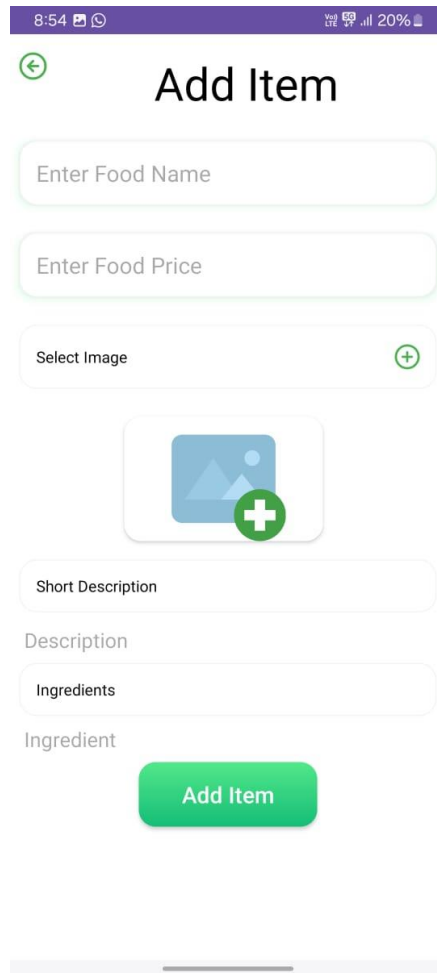
Figure 14 Admin Panel

(Above image shows that the user is accessing the admin panel)

Figure 15 User is adding Food Item

(Above image shows that the user is adding the food item in the main menu)

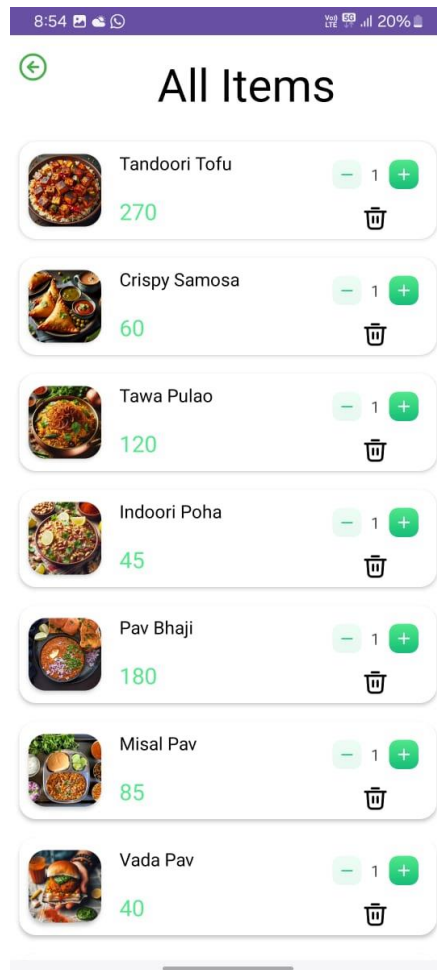*Figure 16 User Viewing the item page*

(From the above image we can see that the user is currently seeing the current added items page)
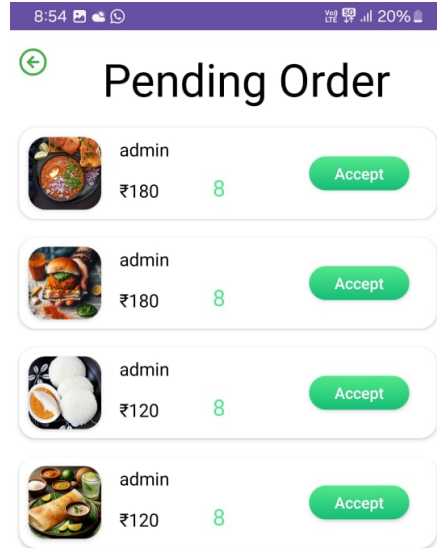
Figure 17 Pending Order Screen

(From the above the screen we can see that the user is seeing the pending order screen)
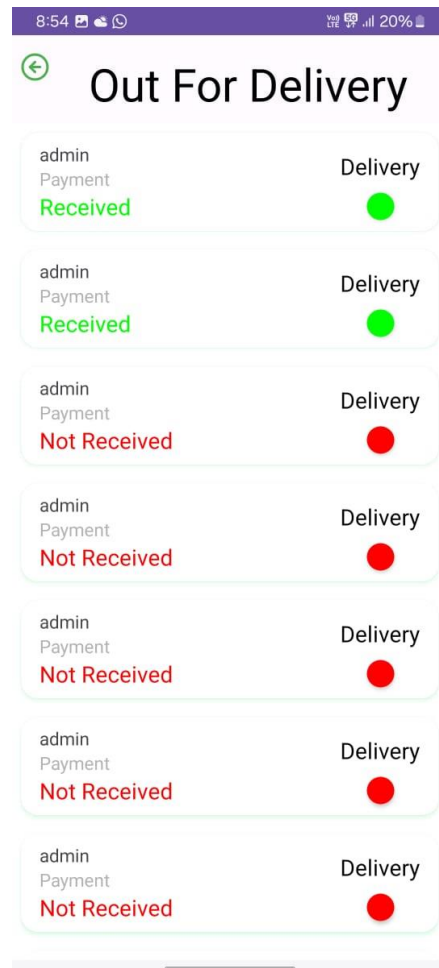
Figure 18 Order Out for delivery

(From the above image we can see that user is seeing the out for delivery screen)

CHAPTER 6: FUTURE ENCHANCEMENT

Future Enhancement in Quick Eat

- **Integration with Loyalty Programs**: Enable users to link their loyalty accounts with participating restaurants through the app, allowing them to earn and redeem rewards seamlessly.
- **Augmented Reality Menu Visualization**: Implement augmented reality features that enable users to visualize menu items in 3D or AR, enhancing the dining experience and helping users make informed choices.
- **Voice-Activated Assistance**: Introduce voice-activated commands and virtual assistants to the app, allowing users to perform tasks such as finding restaurants, browsing menus, and making reservations hands-free.
- **Allergen and Dietary Information**: Expand the app's capabilities to include detailed allergen and dietary information for menu items, catering to users with specific dietary restrictions or preferences.
- **Table Booking and Seating Preferences**: Enhance the reservation feature to allow users to specify seating preferences (e.g., window seat, outdoor patio) and book specific tables at restaurants.
- **Food Delivery Drone Integration**: Partner with food delivery services to offer drone delivery options for users, providing faster and more efficient delivery of orders.
- **Virtual Dining Experiences**: Introduce virtual dining experiences that allow users to virtually explore restaurant interiors, view chef demonstrations, and attend virtual tasting events.
- **Real-Time Waitlist Updates: Implement real-time waitlist updates for restaurants, allowing users** to join virtual waitlists and receive notifications when their table is ready.
- **Food Sustainability Metrics**: Integrate sustainability metrics into the app, providing information about restaurants' sustainability practices, sourcing of ingredients, and environmental impact.
- **Culinary Events and Classes**: Offer users the opportunity to discover and participate in culinary events, cooking classes, and food-related workshops hosted by restaurants or local chefs.

Optimization of the present code

1. **Use Kotlin's Features Efficiently**: Leverage Kotlin's concise syntax and features like data classes, smart casts, and extension functions to write cleaner and more efficient code.

2. **Minimize Object Creation**: Reduce unnecessary object creation, especially within loops, by reusing objects or using Kotlin's apply or run functions to configure objects.

3. **Avoid String Concatenation**: Use StringBuilder for string concatenation in performance-critical sections instead of using the + operator repeatedly.

4. **Optimize Data Structures:** Choose the right data structures for your needs. Consider using sets, maps, or sparse arrays when appropriate to reduce memory consumption and improve lookup times.

5. **Reduce Method Calls:** Minimize the number of methods calls within loops or critical sections to reduce overhead.

6. **Avoid Nested Loops**: If possible, avoid deeply nested loops, as they can lead to exponential time complexity.

7. **Keep Dependencies Updated:** Regularly update third-party libraries and dependencies to benefit from bug fixes, performance improvements, and new features.

8. **Properly Handle Resources**: Ensure that resources like files, database connections, and network sockets are properly closed and released when they are no longer needed.

CHAPTER 7: CONCLUSION

Conclusion

In conclusion, the restaurant-focused application presents a valuable solution for users seeking convenient and enjoyable dining experiences. By providing comprehensive restaurant information, menu browsing capabilities, reservation and ordering functionalities, and user interaction features, the application enhances the way users discover, explore, and engage with restaurants. With the potential for future enhancements such as loyalty program integration, augmented reality menu visualization, and sustainability metrics, the application is poised to further elevate the dining experience and meet the evolving needs of its user base. As technology continues to advance and user preferences evolve, the application stands as a testament to the possibilities of digital innovation in enhancing the way we dine out.

CHAPTER 8: BIBLIOGRAPHY

Reference Links

- https://kotlinlang.org/docs/home.html

- https://developer.android.com/kotlin

- https://www.w3schools.com/KOTLIN/index.php

- https://www.geeksforgeeks.org/kotlin-programming-language/

Video Links

- https://youtu.be/EExSSotojVI?si=GZTSVKgi1HH9RIng

- https://youtu.be/KAh2TOrtTq4?si=BtuBXbZZ_MmNDhM7

- https://youtu.be/ggSP_fy-e-A?si=dtLAFw5KrRjqcL1Z

- https://youtu.be/FlBhpm9aRUg?si=JhevKy7kL_-lYe5P