A Project Report On

Coffee Shop

Submitted in partial fulfillment of the requirement for the

award of the degree

Master of Computer Applications
(MCA)

Academic Year 2025 – 26

Viraj Jethva 92400584040
Harshit Varsani 92400584039
Yash Nageshree 92400584185

Internal Guide
Dr. Ashwin Dobariya

Faculty of Computer Applications (FoCA)

Certificate

This is to certify that the project work entitled

Coffee Shop

submitted in partial fulfillment of the requirement for

the award of the degree of

Master of Computer Applications (MCA)
of the

Marwadi University

is a result of the bonafide work carried out by

Viraj Jethva 92400584040

Harshit Varsani 92400584039

Yash Nageshree 92400584185

during the academic year 2025-26

Faculty Guide

HOD

Dean

DECLARATION

We hereby declare that this project work entitled Coffee Shop is a record done by us.

We also declare that the matter embodied in this project is genuine work done by us and has not been submitted whether to this University or to any other University / Institute for the fulfillment of the requirement of any course of study.

Place:

Date:

Viraj Jethva 92400584040   Signature: _____
Harshit Varsani 92400584039   Signature: _____
Yash Nageshree 92400584185 Signature: _____

ACKNOWLEDGEMENT

It is indeed a great pleasure to express our thanks and gratitude to all those who helped us. No serious and lasting achievement or success one can ever achieve without the help of friendly guidance and co-operation of so many people involved in the work.

We are very thankful to our guide **Dr. Ashwin Dobariya,** the person who makes us to follow the right steps during our project work. We express our deep sense of gratitude to for his /her guidance, suggestions and expertise at every stage. A part from that his/her valuable and expertise suggestion during documentation of our report indeed help us a lot.

Thanks to our friend and colleague who have been a source of inspiration and motivation that helped to us during our project work.

We are heartily thankful to the Dean of our department **Dr. R. Sridaran** sir and HoD **Dr. Sunil Bajeja** sir for giving us an opportunity to work over this project and for their end-less and great support to all other people who directly or indirectly supported and help us to fulfil our task.

Viraj Jethva 92400584040   Signature: _____
Harshit Varsani 92400584039   Signature: _____
Yash Nageshree 92400584185 Signature: _____

CONTENTS

Table Index

Figure index

1) <u>Introduction to Project Definition</u>

The Coffee Shop Application is designed to provide a digital platform for managing and ordering coffee in a simple, fast, and organized way. In today's world, where time is valuable, people prefer easy solutions for daily needs such as food and beverages. Traditional methods of ordering in a coffee shop often take more time and may lead to mistakes in billing or serving. To solve these problems, this project introduces an application that makes the whole process easier for both customers and the shop.

This project focuses on creating an application where the menu of coffees is available in a structured manner. Users can see the different types of coffees, their prices, and details in one place. The system allows customers to select their choices, calculate the total price automatically, and move forward with the order in a smooth way. The design ensures that the entire process is user-friendly, accurate, and quick.

The main aim of this project is to improve efficiency in a coffee shop by reducing manual work, avoiding calculation errors, and saving time. Instead of handling everything with pen and paper, this digital approach ensures better management. It also improves the overall experience because customers can clearly see all available options and make decisions comfortably.

By using this system, the coffee shop becomes more organized and modernized, which is beneficial for both the business and the customers. It shows how technology can make even small daily activities, like buying a cup of coffee, more convenient and enjoyable.

2) <u>Preamble</u>

Coffee shop operations require digital solutions for faster, more efficient, and error-free order processing. This mobile app simplifies customer interactions and empowers the administrator to manage product data in real-time, eliminating manual paper-based ordering and enhancing customer experience.

2.1 Module Description

User Side:
- Login with any username & Password
- Browse coffee items (cappuccino, mocha, flat white, etc.)
- Adjust quantity via plus/minus buttons
- View live total in cart
- Proceed to a payment page

Admin Side:
- Login with any username & Password
- Add new coffee items with name and price.
- Edit item prices or names.
- Delete old or wrong items.
- Changes reflect live in the user menu.

3) <u>REVIEW OF LITERATURE</u>

Similar Systems Studied:
  ➢ **Starbucks App: -**
  ➢ Starbucks mobile app allows users to order drinks, customize them, and pay via the app.
  ➢ Menu is dynamic and managed by the admin side from the server.

Comparison: Your app has similar basic Starbucks ordering but on a local level using SQLite instead of cloud.

Conclusion from Review:
  - Your Coffee Ordering App implements many important features from leading systems.
  - While large apps are complex and online, your app is simpler, faster, and suitable for local cafes or college projects.
  - It covers all basic functionality like menu display, cart, admin management, and payment simulation — all in one place.

4) <u>Technical Description</u>

4.1) <u>Hardware Requirement</u>

Component	Specification
Android Device	Minimum Android Version: 5.1
	Recommended RAM**:** 2 GB or more
	Screen Size: 5 inches or more (for UI clarity)
Computer/Laptop	Minimum 4 GB RAM**,** i3 Processor
	Disk Space: at least 2 GB free for project
USB Cable	To run the app on a physical Android device

Table 4.1 Hardware Configuration

4.2) <u>Software Requirement</u>

Software	Purpose
Android Studio	IDE to build the app using Java and XML
Java JDK 8 or above	Required to compile Java code
SQLite (built-in)	To store and manage menu items locally
Gradle	Build tool used by Android Studio
Android Emulator	To test the app without a real device

Table 4.1 Software Requirement

5) <u>System Design and Development</u>

5.1 <u>Algorithm</u>

➢ **User Module Algorithm**

Step 1: Start the app, Display homepage with User Login and Admin Login buttons

Step 2: User clicks User Login

Step 3: Enter any username, Password Click "Login"

Step 4: Open menu screen

Step 5: Load items from SQLite database

Step 6: For each item in menu:

Show name, price, and quantity counter

User clicks + button – increase quantity

User clicks – button – decrease quantity (if > 0)

Step 7: On "Go to Cart" click:

Display only selected items

Calculate total price = sum of (item price × quantity)

Step 8: User selects Cash or UPI (optional future scope)

Step 9: Click on Confirm Payment – Display "Payment Successful" toast/message

Step 10: End

➢ Admin Module Algorithm

Step 1: Admin clicks Admin Login

Step 2: Enter fixed credentials: username, Password Click "Login"

Step 3: If login valid - go to Admin Panel

Step 4: Admin can perform:

Add Item - insert into SQLite table

Update Item - update name/price in SQLite

Delete Item - remove item from SQLite

All items are shown in ListView with click-to-edit support
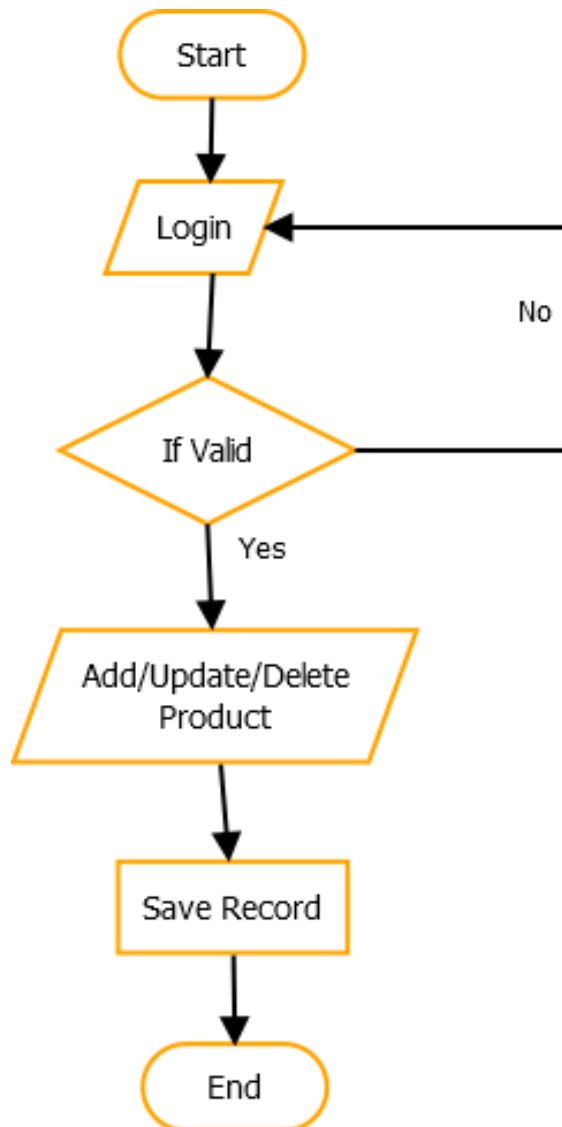
Step 5: End

<u>Flowchart: -</u>

Admin Flowchart

Figure 5.2 Admin Flow Chart

User Flowchart
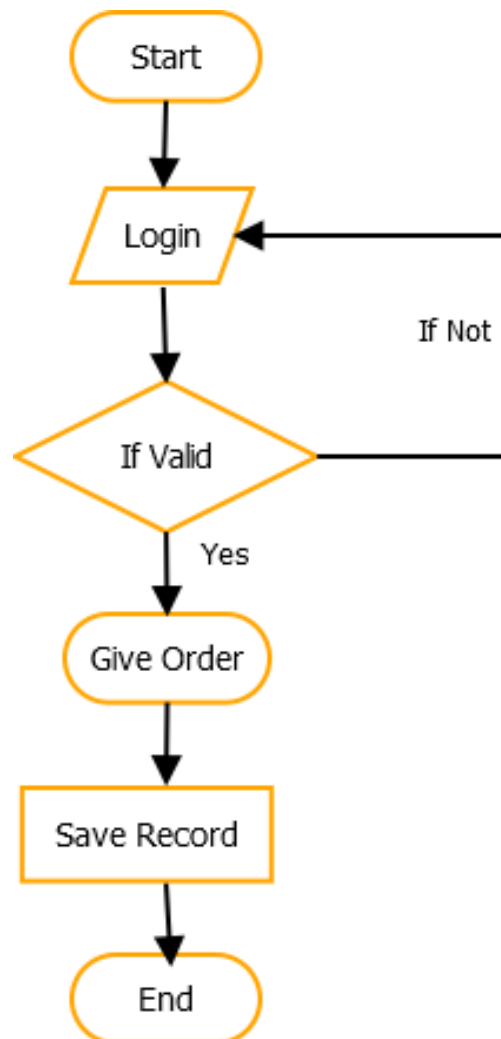
Figure 5.3 Admin Flow Chart

Data Flow Diagram
(Zero Level)

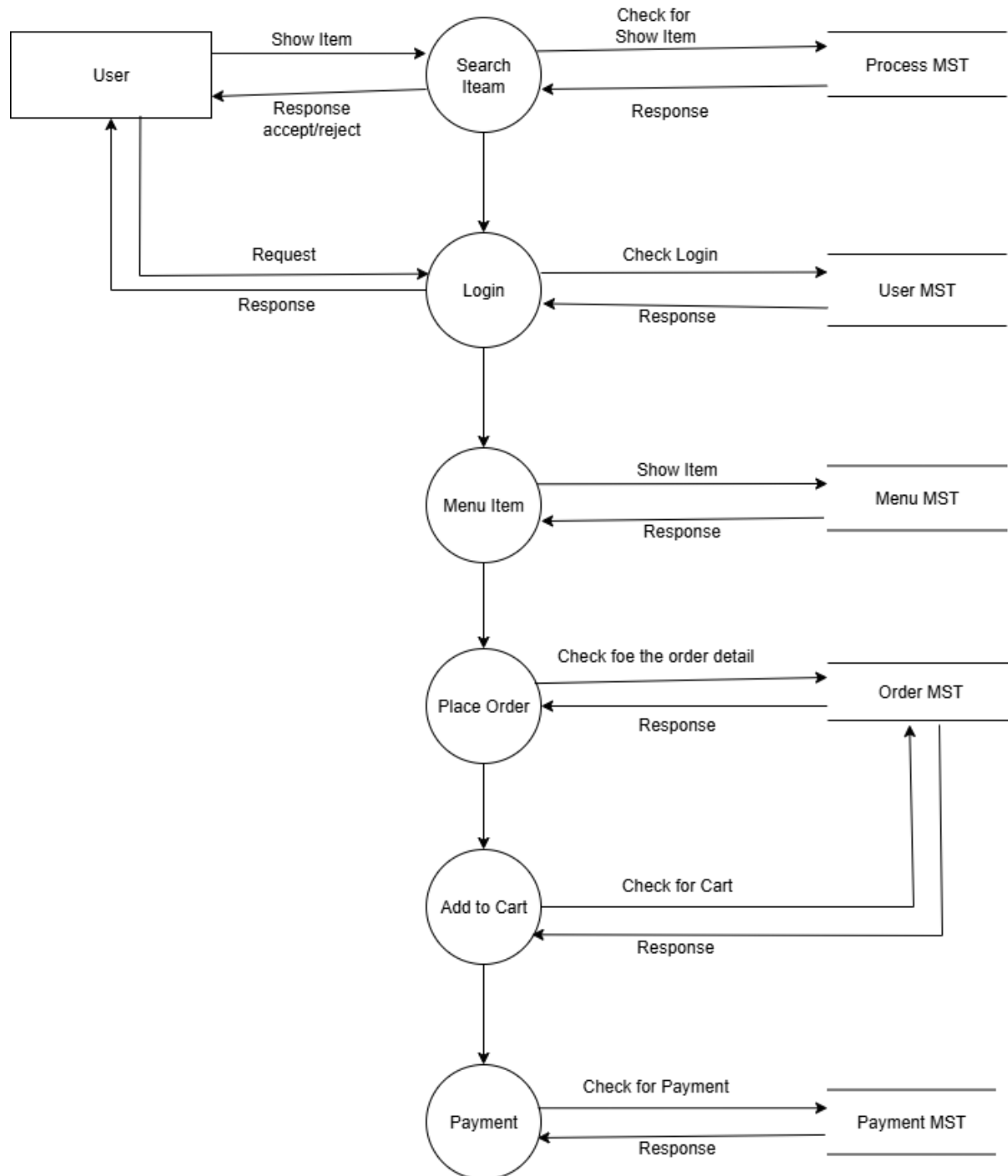# Figure 5.4 Data Flow Diagram (Level 0)

Data Flow Diagram
(First Level)

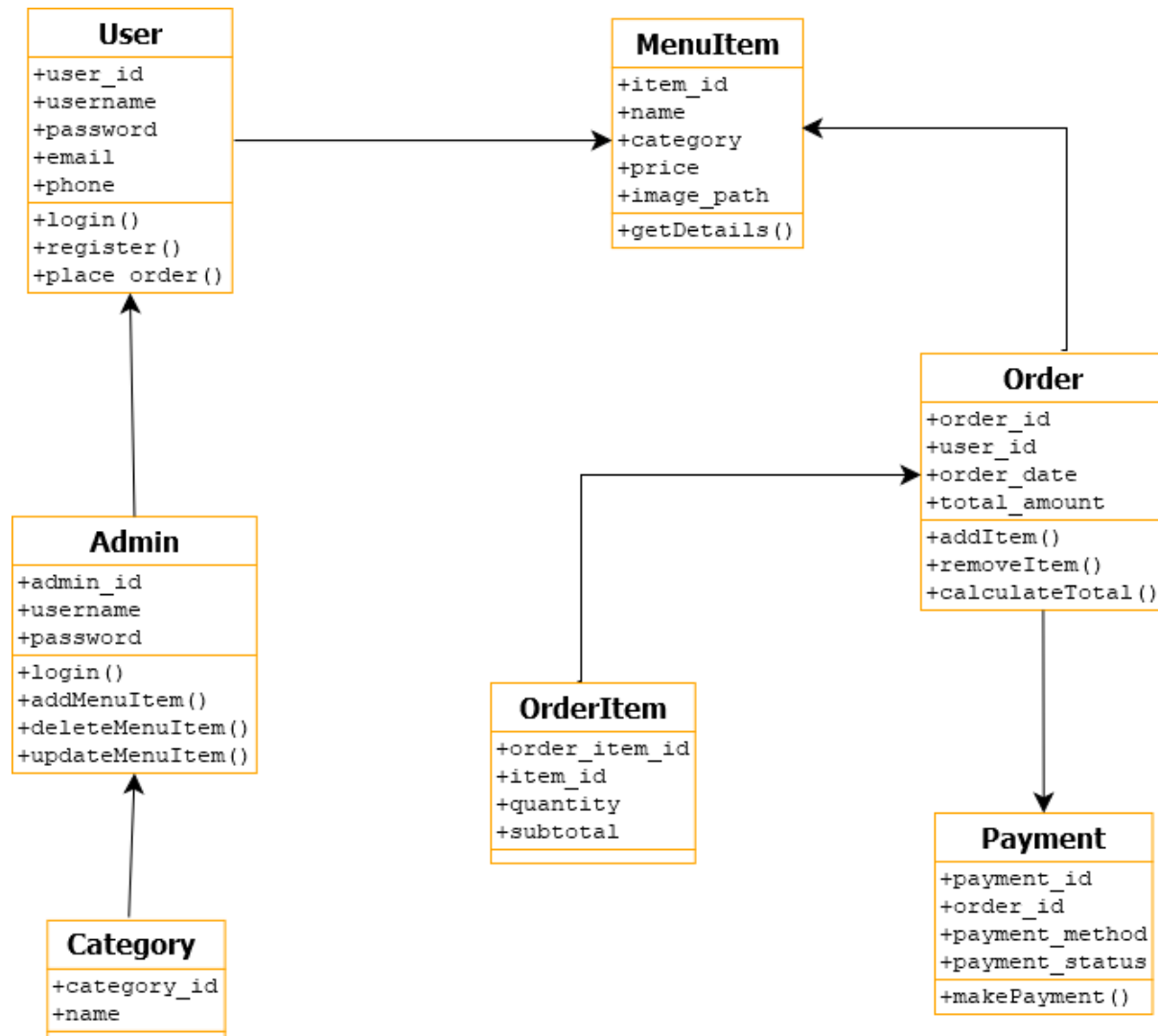## Figure 5.5 Data Flow Diagram (Level 1)

Class Diagram: -

User
+user_id
+username
+password
+email
+phone
+login()
+register()
+place order()

MenuItem
+item_id
+name
+category
+price
+image_path
+getDetails()

Admin
+admin_id
+username
+password
+login()
+addMenuItem()
+deleteMenuItem()
+updateMenuItem()

Order
+order_id
+user_id
+order_date
+total_amount
+addItem()
+removeItem()
+calculateTotal()

OrderItem
+order_item_id
+item_id
+quantity
+subtotal

Category
+category_id
+name

Payment
+payment_id
+order_id
+payment_method
+payment_status
+makePayment()

Figure 5.6 Class Diagram

<u>Use Case Diagram: -</u>
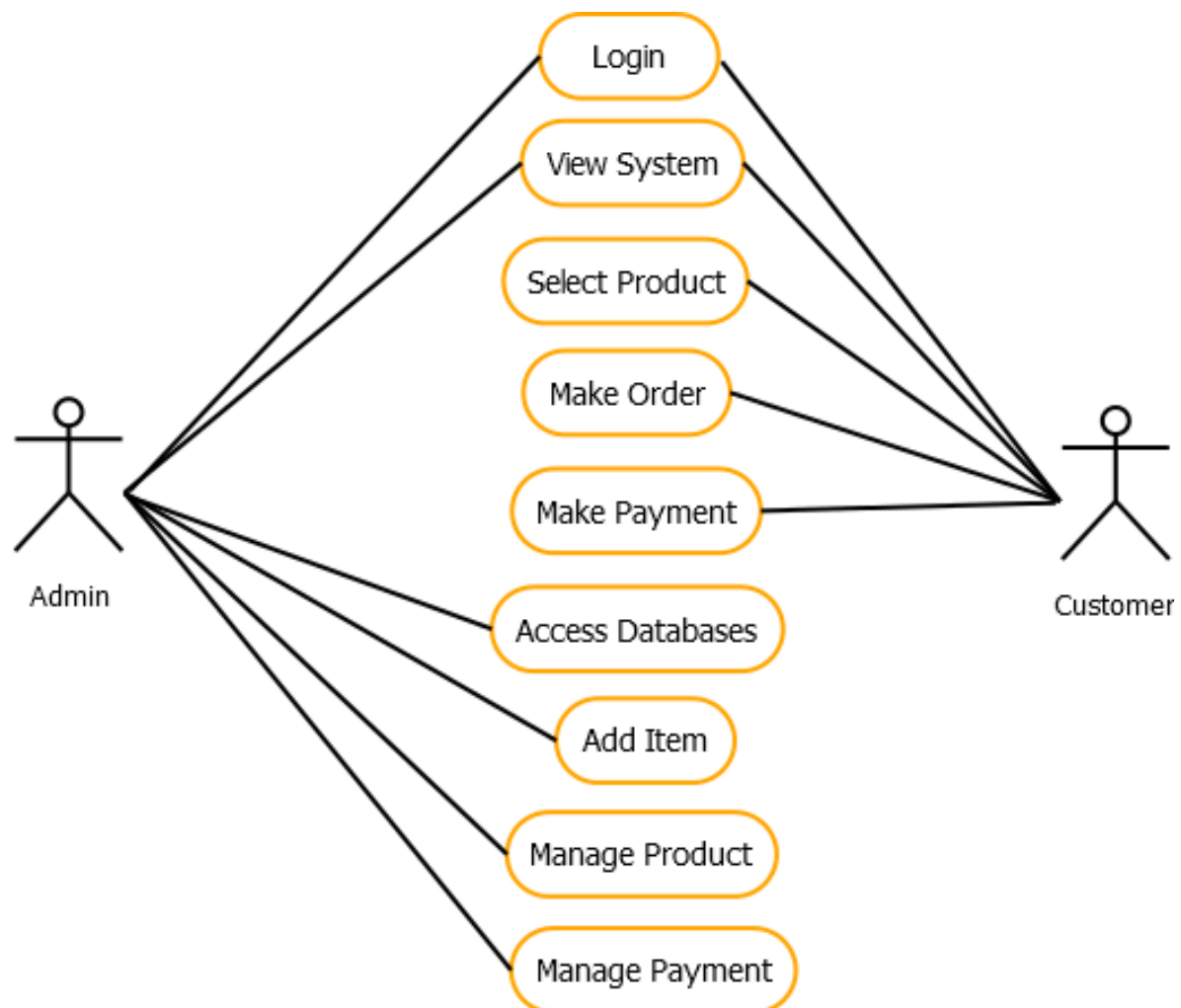
Figure 5.7 Use Case Diagram
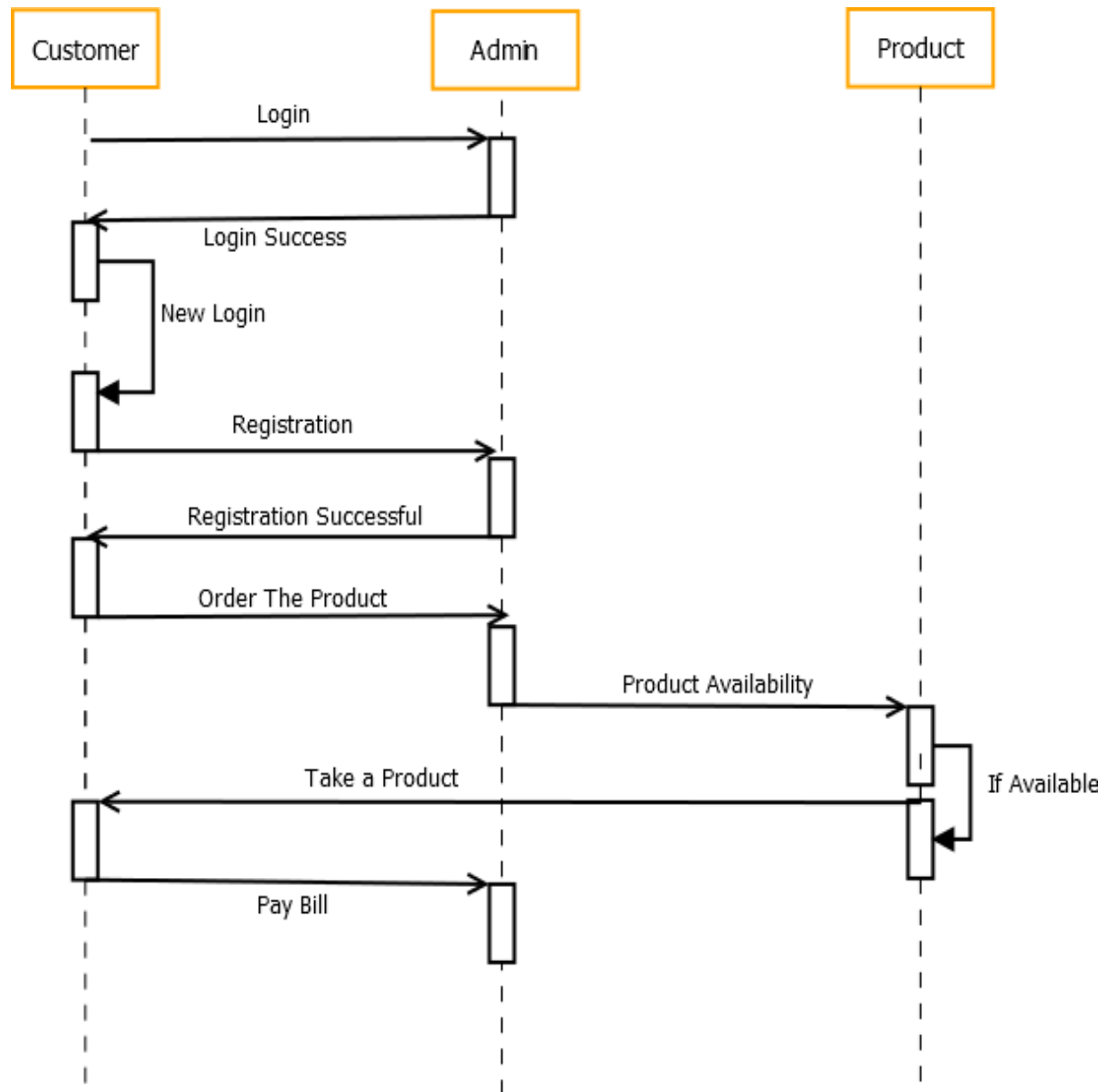
Sequential Diagram: -
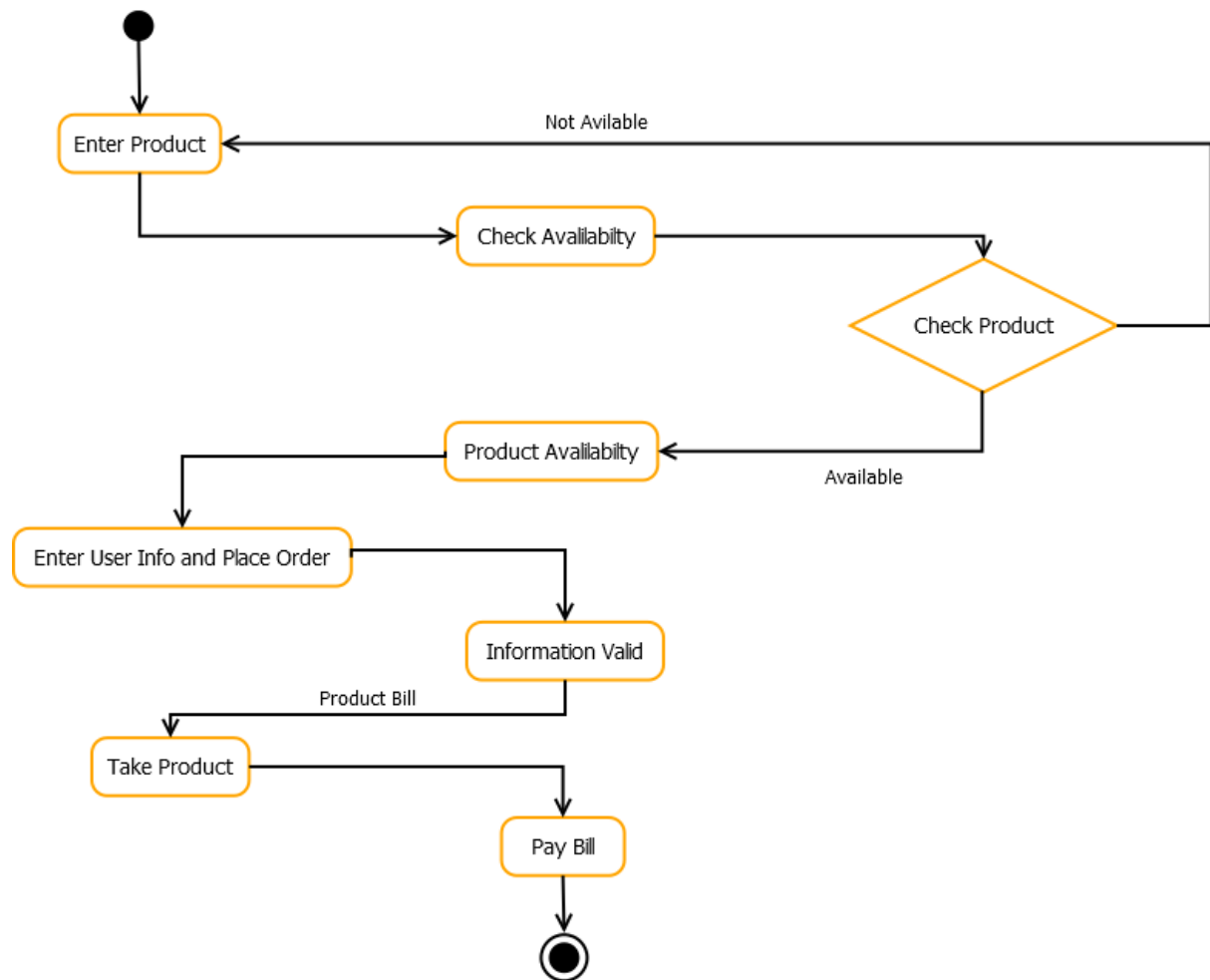

Figure 5.8 Sequential Diagram

Activity Diagram

Figure 5.9 Activity Diagram

COFFEE SHOP

State Diagram: -
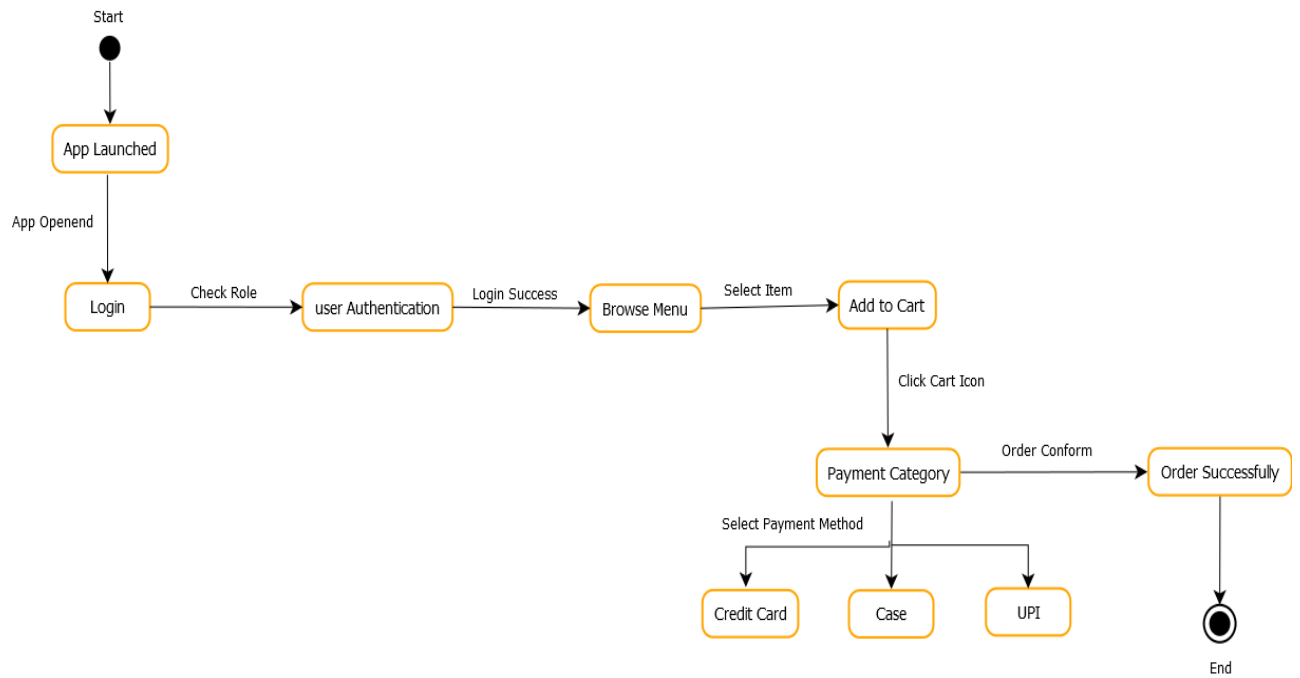

Figure 5.10 State Diagram

Database Table:

Table Name: users

Attribute Name	Datatype	Constraints
Id	Int	Primary Key
Name	Varchar (10)	-
Email	Varchar (10)	-
Password	Int	-

Table 5.11 Users Table Structure

Table Name: menu_item

Attribute Name	Datatype	Constraints
Id	Int	Primary Key
Category_id	Int	-
Name	Varchar (20)	-
Price	int	-

Table 5.12 Menu_Item Table Structure

Table Name: orders

Attribute Name	Datatype	Constraints
Id	Int	Primary Key
User_id	int	-
Item name	Varchar (100)	-
Quantity	Int	-

Table 5.13 Orders Table Structure

Table Name: category_table

Attribute Name	Datatype	Constraints
Id	Int	Primary Key
Name	Varchar (20)	-

Table 5.14 Category_Table Structure

Table Name: order_status

Attribute Name	Datatype	Constraints
Id	int	Primary Key
User_id	int	-
total	int	-
status	Varchar (10)	-

Table 5.15 Order_Status Table Structure

Table Name: admin

Attribute Name	Datatype	Constraints
Id	Int	Primary Key
Username	Varchar (50)	-
Password	Varchar (50)	-

Table 5.16 Admin Table Structure

Screen Design: -

User Login: -
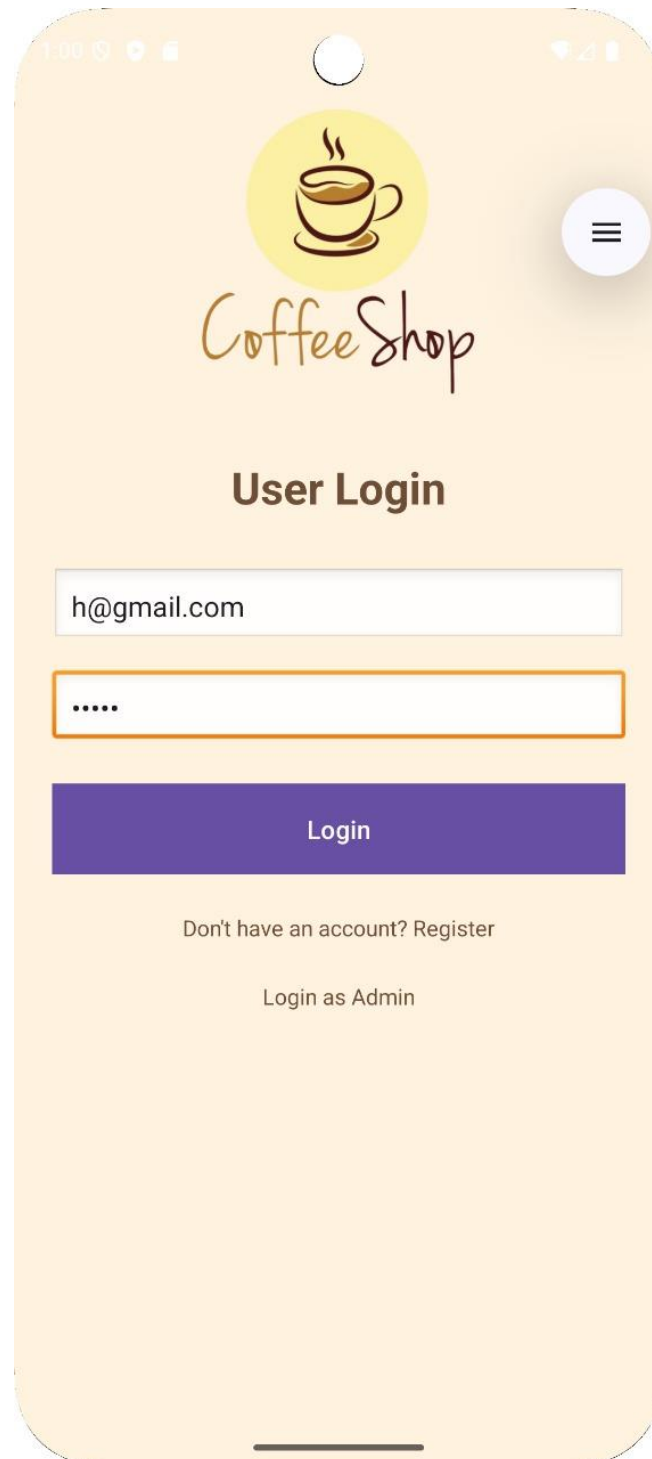
Figure 5.17 User Login Screen

In this page we will display the user login with Email and Password to login the app of the Coffee Shop
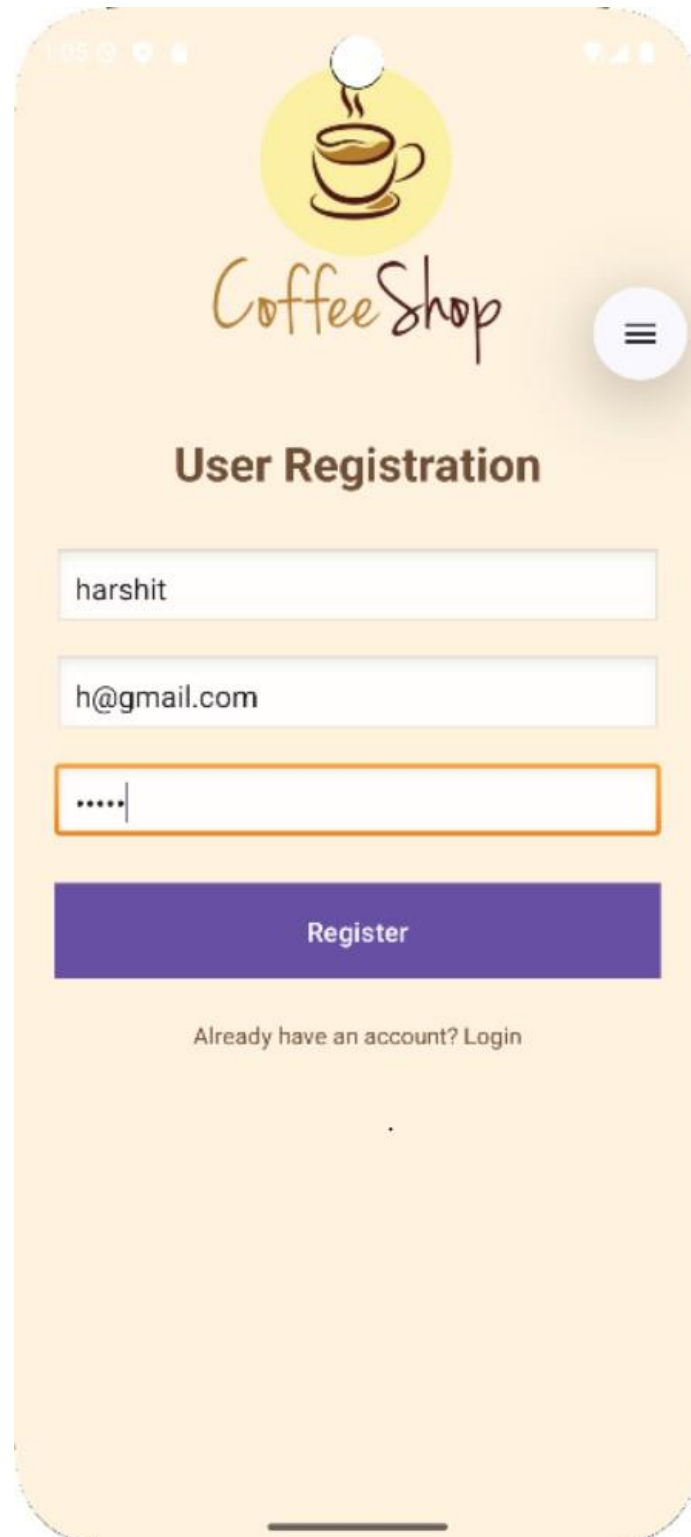
Splash Screen: -

Figure 5.18 Splash Screen

In this screen we will display the splash activity of the after the login is done the logo will display for 3 seconds and menu will display after that

<u>User Registration: -</u>

Figure 5.19 User Registration Screen

In this screen we will display the user registration with name, email, password is showing. User will register in the app

COFFEE SHOP

Menu Item: -

Figure 5.20 Menu Screen

In this screen we will display the all-coffee menu items with scrollable view form
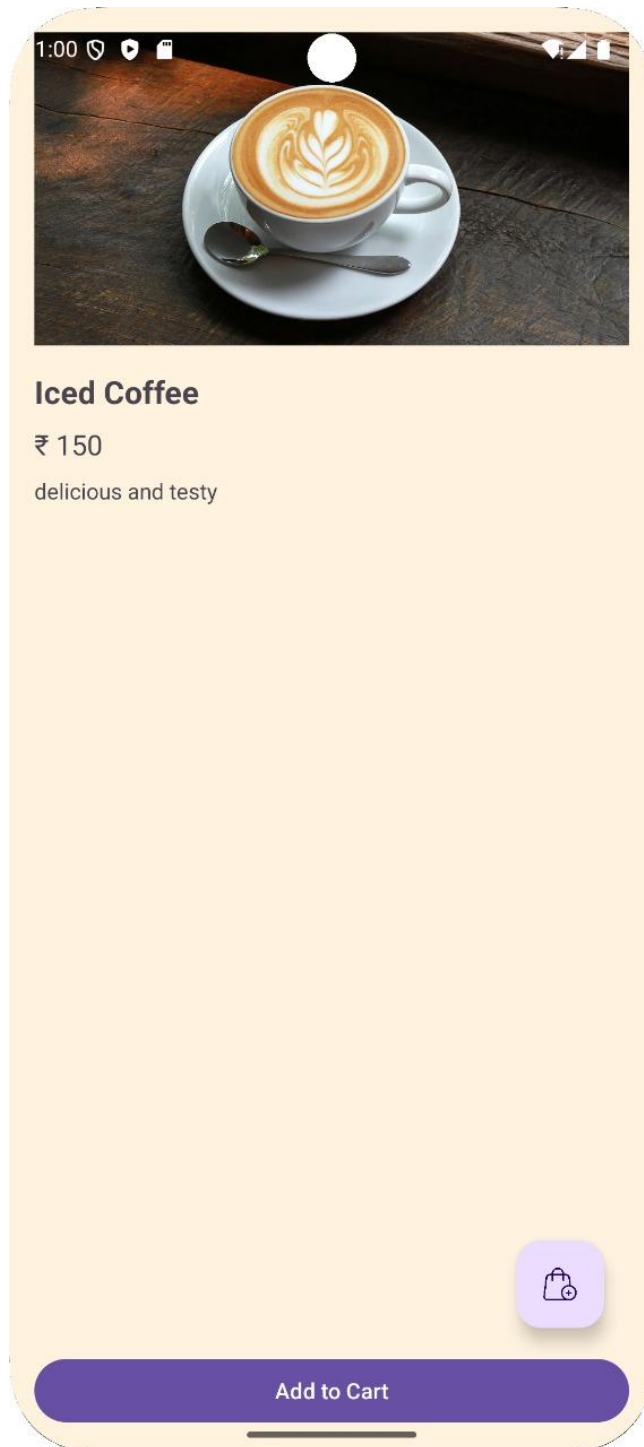
<u>Item Detail: -</u>

Figure 5.21 Item Detail Screen

In this screen we display the menu items are showing in the screen with their price and quantity

Your Cart: -

Figure 5.22 Cart Screen

In this screen we will showing the add items with their price and quantity
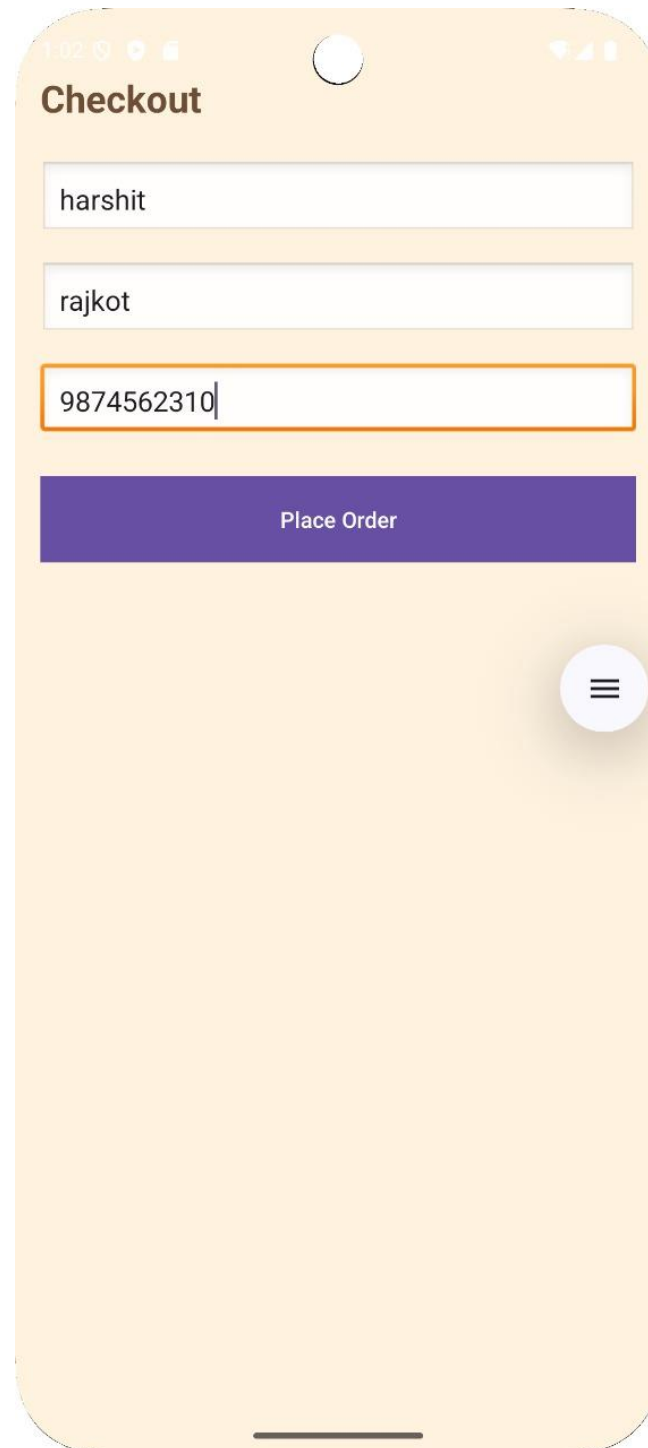
Check Out menu: -

Figure 5.23 Checkout Screen

In this screen we will showing the display user detail like name, address, phone number and conform click on place order button

Make a Payment: -

Figure 5.24 Payment Screen

In this screen we will display the payment categories and choose any one and click on processed to pay button on it

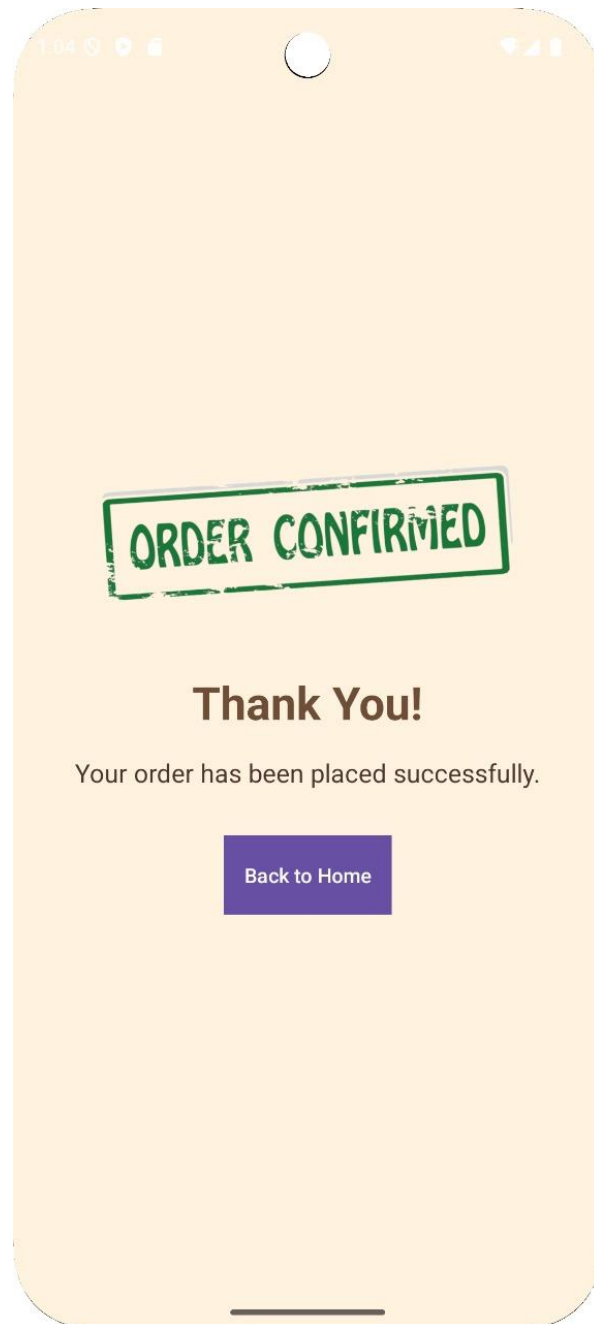## <u>Order Conformation: -</u>

Figure 5.25 Order Confirmation Screen

This is the final page of the app will show the order will place successfully and order conformed will display and click back to home again
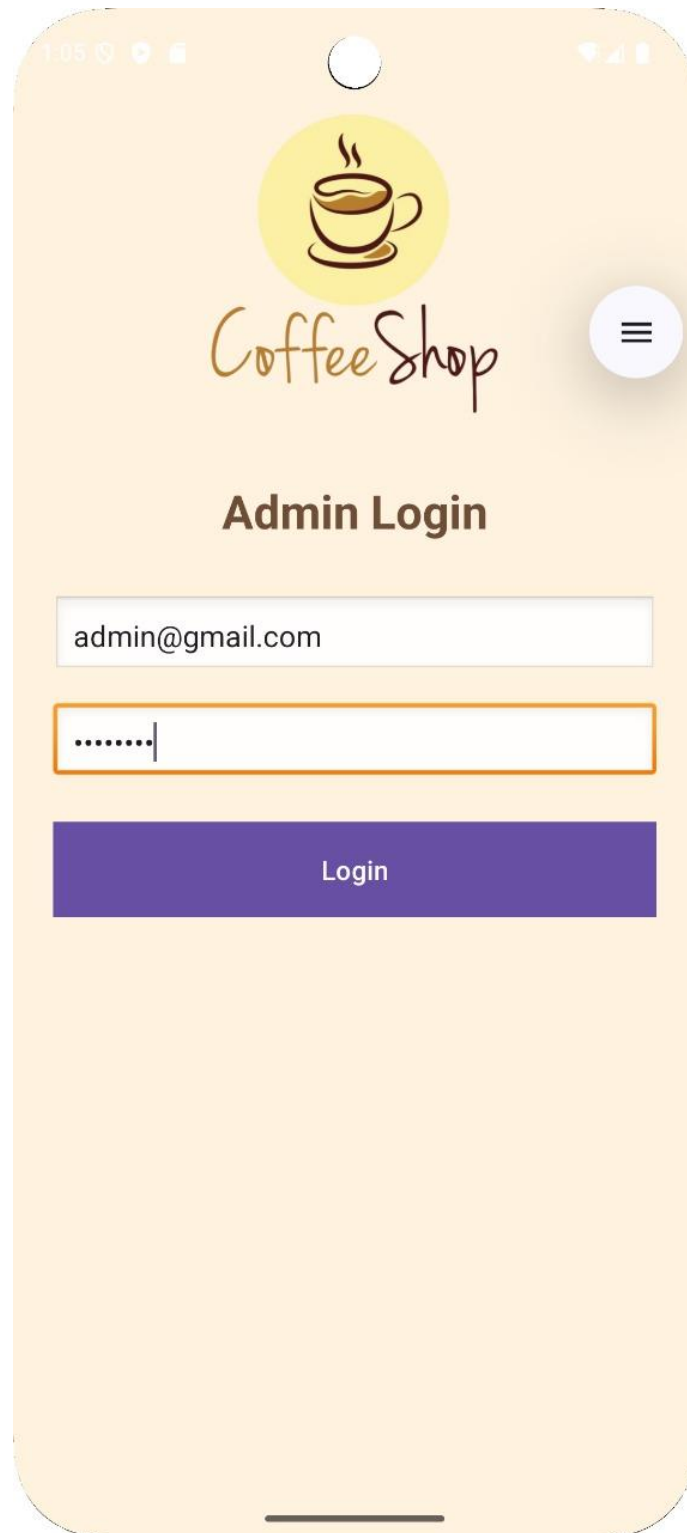
Admin login: -

Figure 5.26 Admin Login Screen

In this we will display the admin login page with their email, and password and click on login button
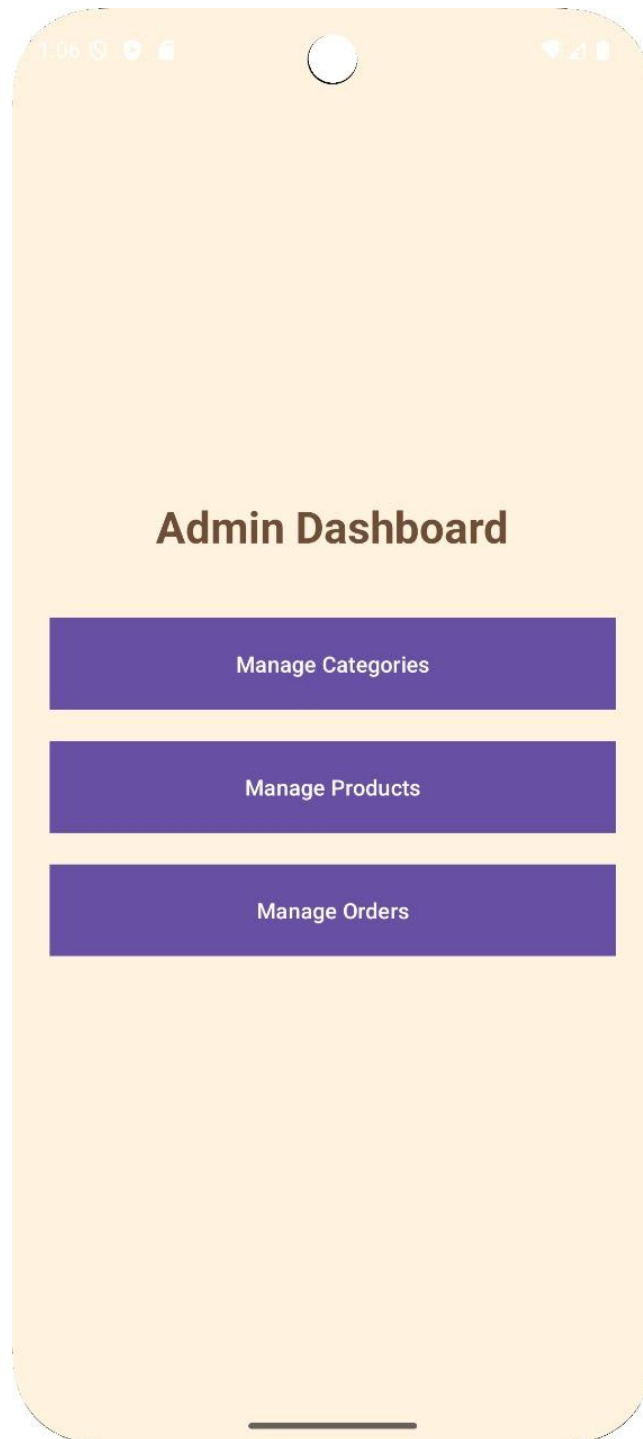
<u>Admin Dashboard: -</u>

Figure 5.27 Admin Dashboard Screen

In this screen we will display admin dashboard categories to manage the things like add, update and delete and manage menu list

<u>Add item Page: -</u>

Figure 5.28 Add Item Page Screen

In this page we will show the product name, price, category etc. to update the things in menu and click on save button

6) <u>System Testing:</u>

6.1 Objective:

System testing ensures that the Coffee Shop Application (with menu, add-to-cart, order, etc.) works as a complete and integrated system according to requirements.

6.2 Scope:

The testing will cover:

- **Functional Testing** – Verifying all features (menu display, add/remove items, order confirmation, payment).
- **UI Testing** – Buttons, layouts, navigation flow.
- **Performance Testing** – App response with multiple items/orders.
- **Compatibility Testing** – Runs on different Android versions/devices.
- **Error Handling** – Invalid inputs, missing selections.

6.3 Features to be Tested:

Feature	Description	Expected Result
App Launch	Open Coffee Shop app	Splash screen → Homepage
Menu Display	Show coffee items with price	All items visible
Add Item (+)	Add coffee item to cart	Item count increases
Remove Item (-)	Remove item from cart	Item count decreases
Cart Summary	Shows selected items & total	Correct total displayed
Place Order	User confirms order	payment screen
Error Handling	Try order with empty cart	Show error message
Exit App	User closes app	App exits gracefully

Table 6.3 Features to be Tested

COFFEE SHOP

6.4 Test Cases:
<u>Functional Test Cases</u>

Test Case ID	Test Case Description	Test Steps	Input Data	Expected Result	Actual Result	Pass/Fail
TC-01	Verify app launch	1. Open app	N/A	Splash screen → Homepage	Homepage displayed correctly	Pass
TC-02	Add item to cart	1. Select Cappuccino 2. Click (+)	Cappuccino	Item added to cart, qty=1	Item added, qty=1	Pass
TC-03	Remove item from cart	1. Add Cappuccino 2. Click (–)	Cappuccino	Qty decreases / item removed	Item not removed, qty stays same	Fail
TC-04	Add multiple items	1. Add Latte 2. Add Espresso 3. Add Mocha	Latte, Espresso, Mocha	All items visible in cart	Only Latte and Espresso visible (Mocha missing)	Fail
TC-05	Empty cart order	1. Open cart 2. Click Place Order	No items	Error message "Cart Empty"	App crashes unexpectedly	Fail
TC-06	Place order with items	1. Add Cappuccino (₹200), Latte (₹180) 2. Click Place Order	2 items	Bill screen displayed with total ₹380	Bill screen displayed with wrong total ₹360	Fail
TC-07	Exit app gracefully	1.Press back button twice	N/A	App exits Without crash	App exited properly	Pass

Table 6.4 Functional Test Cases

6.5 Tools for Testing:
- Android Studio Emulator
- Physical Device Testing

6.6 Exit Criteria:
- All critical test cases passed (Add/Remove items, Place Order,)
- UI works across multiple devices.

7) <u>Conclusion:</u>

This Coffee Shop app project allows users to browse different coffee items, add them to a cart, and make orders easily. It demonstrates basic app functionalities like user login, admin login, database connection, item management, and a smooth user interface. Overall, the project shows how an online ordering system works in a small business, providing convenience to customers and helping shop owners manage orders efficiently.

8) <u>Learning During Project Work:</u>

- Learned how to design a user-friendly app interface.
- Gained knowledge of Java and XML for Android development.
- Understood how to connect an app with a database (MySQL).
- Learned to implement features like login, cart, and payment.
- Improved problem-solving and debugging skills.

8.1) <u>Future Enhancement:</u>

- Add online payment integration (like PayPal or UPI).
- Include more coffee varieties and offers.
- Enable order tracking in real-time.
- Add user feedback and rating system.
- Make the app available on multiple platforms (iOS, Web).

9) <u>Bibliography:</u>

9.1) Online References:

[1] Tutorials Point, "Android SQLite Database Tutorial." [Online]. Available: https://www.tutorialspoint.com/android/android_sqlite_databa se.htm. [Accessed: Jun. 5, 2025].

[2] GeeksforGeeks, "SQLite in Android." [Online]. Available: https://www.geeksforgeeks.org/sqlite-in-android/. [Accessed: July.16, 2025].

[3] MongoDB Inc., "What is MongoDB?" [Online]. Available: https://www.mongodb.com/docs/manual/introduction. [Accessed: July. 25, 2025].

9.2) Offline References:

[1] A. Silberschatz, H. F. Korth, and S. Sudarshan, Database System Concepts, 6th ed. New York, NY, USA: McGraw-Hill, 2019.

[2] R. Elmasri and S. B. Navathe, Fundamentals of Database Systems, 7th ed. Boston, MA, USA: Pearson, 2016.

[3] H. Schildt, Java: The Complete Reference, 11th ed. New Delhi, India: McGraw-Hill, 2019.