ICBT CAMPUS

Where Life Begins...

Cardiff Metropolitan University | Prifysgol Metropolitan Caerdydd

Mobile Application and Development

Warun Aloka

CL/HDCSE/CMU/117/94

Contents

Acknowledgment

I am deeply grateful to Mr. Tharaka for the unwavering support and invaluable guidance provided during the mobile application development course. His deep knowledge and commitment to teaching have been instrumental in helping me grasp complex concepts and develop practical skills in mobile app development.

Throughout the course, Mr. Tharaka has shown remarkable dedication, offering not only detailed explanations but also constructive feedback that has significantly improved my work. His enthusiasm for the subject has been truly inspiring, motivating me to push my boundaries and achieve more than I thought possible.

I would also like to extend my sincere thanks to Mrs. Upeka, whose exceptional mentorship has played a crucial role in both my academic and personal growth. Her approach to teaching has fostered a positive and enriching learning environment, making my journey in this course both enjoyable and rewarding.

The knowledge and confidence I have gained from both Mr. Tharaka and Mrs. Upeka have prepared me to face new challenges with enthusiasm and resilience. For this, I am profoundly appreciative.

Task 1

Comprehensive Analysis of Development Tools, Technologies, and Mobile Operating Systems for "The Dog Food" Mobile Application

Mobile Operating Systems

1.Android

Android is the most widely used mobile operating system globally, with a vast and diverse user base. Its open-source nature allows for extensive customization, making it ideal for creating unique user experiences tailored to specific market segments. Android's dominance in many regions, particularly in emerging markets, ensures a broad reach for "The Dog Food" app, maximizing its potential audience. However, the diversity of devices and Android versions can introduce challenges in maintaining consistent app performance across all platforms. Developers must consider this fragmentation and implement rigorous testing to ensure a smooth and uniform experience for all users.

2.iOS

iOS is renowned for its stringent security protocols, uniform user experience across devices, and a highly lucrative app ecosystem. Apps on iOS often see higher in-app purchase rates and quicker adoption by users due to Apple's strong brand loyalty and ecosystem integration. However, the limited number of devices, while ensuring consistency, can also restrict the flexibility in design and functionality. Additionally, Apple's strict App Store guidelines can pose challenges during the app submission process, potentially leading to delays or additional development iterations to meet all requirements.

Development Tools and Technologies

1.Native Development

Native development, using tools like Xcode for iOS and Android Studio for Android, allows developers to harness the full power of each platform's unique features. This method enables the creation of highly optimized, responsive, and visually cohesive apps that can fully utilize device-specific capabilities such as camera functions, GPS, and more. The trade-off, however, is the increased development time and cost, as maintaining separate codebases for iOS and Android requires more resources. This approach is best suited for apps that demand high performance and deep integration with platform-specific features.

2.Cross-Platform Frameworks

Frameworks such as React Native, Flutter, and Xamarin offer a more cost-effective and time-efficient alternative by allowing developers to write a single codebase that runs on multiple platforms. This approach reduces development overhead and accelerates the deployment process. However, the performance of cross-platform apps may not match that of native apps, and certain platform-specific features might be more challenging to implement or optimize. Despite these limitations, cross-platform frameworks are an excellent choice for apps where time-to-market and budget are critical factors, and where the app does not require intensive use of platform-specific features.

Backend Technologies

1.Backend Architecture

A robust backend is essential for handling the core functionalities of the app, such as order processing, data management, and user authentication. Technologies like AWS Amplify, Firebase, and Node.js provide scalable and reliable backend services that can easily integrate with mobile applications. These technologies offer various features, such as real-time database updates, secure authentication, and cloud storage, ensuring that the app remains responsive and secure as the user base grows. A well-architected backend will support "The Dog Food" app's operational needs, enabling seamless interactions between the user interface and backend systems.

Development Tools

Android Studio:

Android Studio is the premier IDE for Android app development, offering a comprehensive suite of tools that streamline the development process. It integrates seamlessly with the Android SDK, providing developers with powerful tools for designing user interfaces, debugging code, and optimizing app performance. The user-friendly design capabilities of Android Studio are particularly beneficial for creating an intuitive and visually appealing interface for "The Dog Food" app, ensuring a positive experience for both administrators and end-users. The IDE's robust features align perfectly with the bakery's objectives, facilitating the development of a high-quality, responsive app.

Xcode:

Xcode is Apple's official IDE for iOS app development, providing developers with all the tools they need to build apps within the Apple ecosystem. Xcode supports the entire development process, from initial design to deployment on the App Store. Its extensive toolkit includes Interface Builder for designing user interfaces, a powerful debugging environment, and comprehensive testing tools. For "The Dog Food" app, Xcode enables the creation of an app that adheres to Apple's high standards for design and functionality, ensuring a seamless user experience across all iOS devices.

Windows API:

Although less relevant for mobile app development, understanding the Windows API is crucial for developers working within the Windows ecosystem. The Windows API provides the necessary functions and protocols to interact with the underlying system, enabling the development of applications that leverage Windows' capabilities. For developers creating software that must operate in a Windows environment, knowledge of the Windows API is essential for ensuring compatibility and performance.

User Roles in "The Dog Food" Mobile Application:

Admin User: The admin user in "The Dog Food" app has enhanced privileges, allowing them to manage critical operational tasks. They can log in to oversee order processing, verify incoming orders, and manage cupcake categories. This role is crucial for the smooth functioning of the bakery, as it ensures that orders are processed efficiently and that the app's content remains up-to-date. The admin interface must be designed with ease of use in mind, allowing administrators to perform their tasks quickly and accurately.

Member User: Member users are the regular customers of "The Dog Food" app. They can create an account, log in, browse cupcakes by category, place orders, and review their order history. The app's design must focus on providing a personalized and user-friendly experience, making it easy for customers to find and order their favorite cupcakes. The member user interface should be intuitive, responsive, and visually appealing to encourage repeat usage and enhance customer satisfaction.

Conclusion:

After evaluating the various mobile operating systems, development tools, and technologies, it is evident that Android is the most suitable platform for "The Dog Food" app. Android's widespread adoption, extensive user base, and compatibility with a wide range of devices make it the ideal choice for reaching a broad audience. Its open-source nature offers flexibility in distribution and development, while Android Studio's powerful features support the creation of a high-quality, user-friendly app. By leveraging Android's strengths, "The Dog Food" app can effectively meet the needs of both administrators and members, driving customer satisfaction and supporting the bakery's growth.

Task 2

2.1Use case Diagram

Admin

- **Admin Login**

  Allows the admin to access the system. This process includes searching in the database and may extend to handling invalid username/password scenarios.

- **Create Category** and **Manage Category**

  These functions enable the admin to create and manage product categories within the system.

- **Create Item** and **Manage Item**

  The admin can create and manage items, associating them with specific categories and order types.

- **Add Discounts**

  Allows the admin to apply discounts to products or categories.

- **Manage Orders**

  The admin can oversee and manage customer orders.

- **Manage Supplier Product**

  Involves managing products provided by suppliers.

- **Create Supplier Account**

  Allows the admin to create accounts for suppliers within the system.

- **Moderate Feedbacks**

  The admin can review and moderate customer feedback.

- **View Supplier Account**

  The admin has the capability to view supplier account details.

- **View User Account**

  The admin can also view details of user accounts.

- **View Orders**

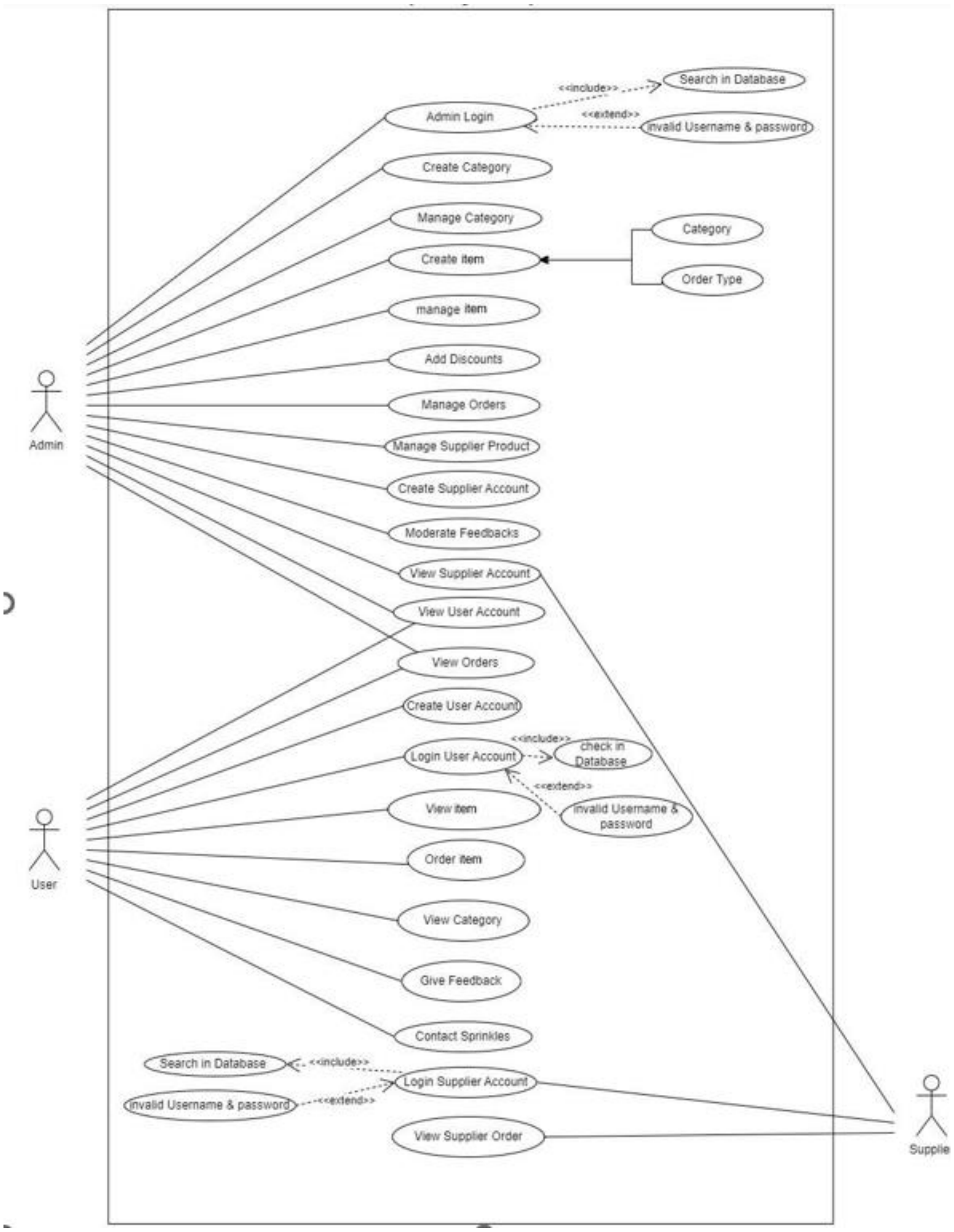
  Enables the admin to view all orders placed by users.

User

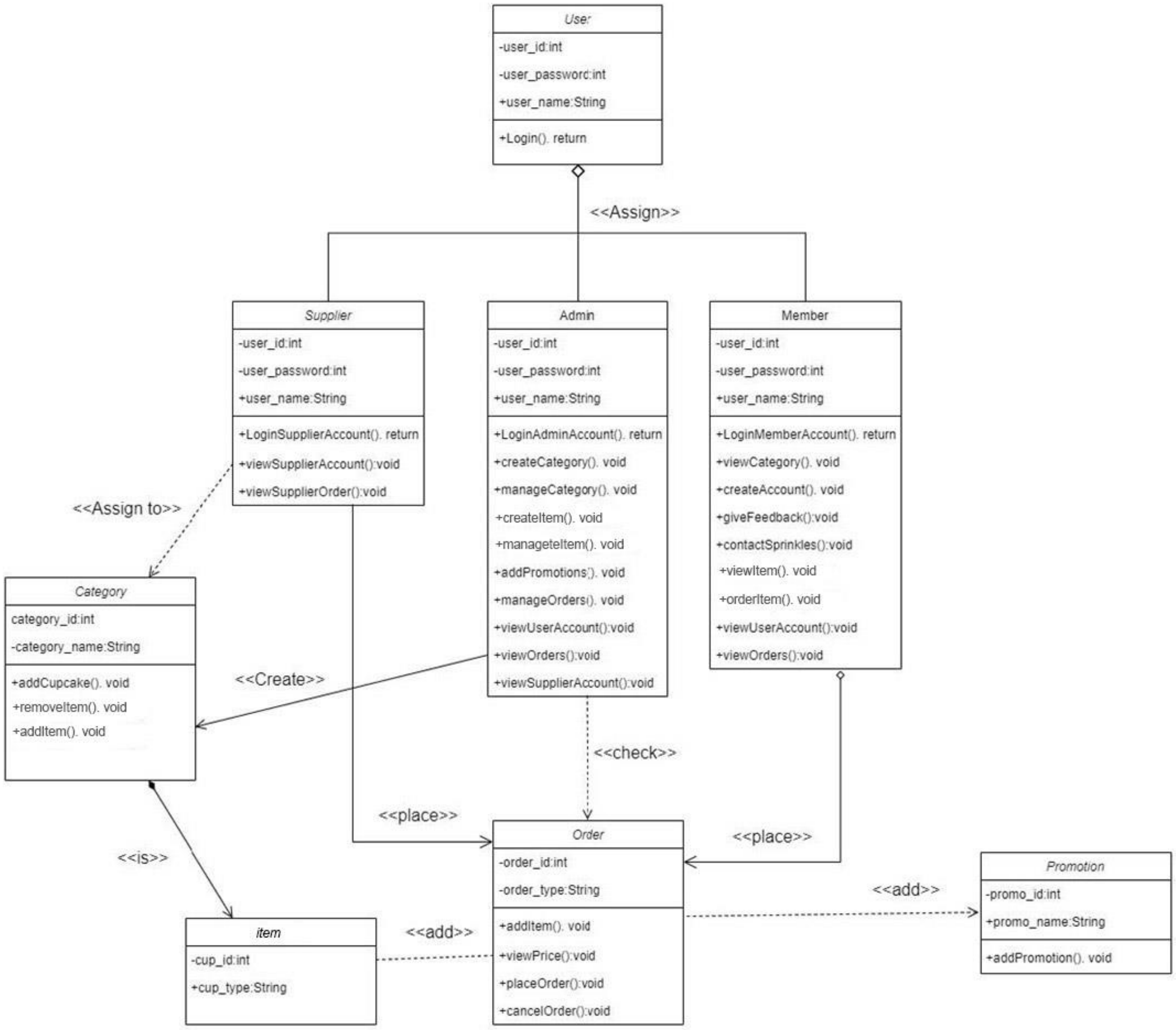
- **Create User Account**: Allows users to create an account to access the system.
- **Login User Account**: Users can log into their accounts, which includes database checks and may extend to managing invalid username/password scenarios.
- **View Item**: Users can browse items available in the bakery.
- **Order Item**: Allows users to place orders for selected items.
- **View Category**: Users can view different product categories.
- **Give Feedback**: Users can provide feedback on products or services.
- **Contact Sprinkles**: Provides users with a way to contact support or customer service.
- **View Orders**: Users can view their order history.

Supplier

- **Login Supplier Account**: Suppliers can log in to their accounts, with the process including database checks and the handling of invalid username/password situations.
- **View Supplier Order**: Suppliers can view orders related to the products they supply.


Admin Login
<<include>>
Search in Database
<<extend>>
invalid Username & password
Create Category
Manage Category
Category
Create item
Order Type
manage item
Add Discounts
Manage Orders
Manage Supplier Product
Create Supplier Account
Moderate Feedbacks
View Supplier Account
View User Account
View Orders
Create User Account
Login User Account
<<include>>
check in Database
<<extend>>
invalid Username & password
View item
Order item
View Category
Give Feedback
Contact Sprinkles
Search in Database
<<include>>
Login Supplier Account
invalid Username & password
<<extend>>
View Supplier Order
Admin
User
Supplie

2.2 Class Diagram



Classes and Attributes

1. **User Class**
   - Attributes: user_id, user_password, user_name
   - Methods: Login()
2. **Admin Class**
   - Attributes: user_id, user_password, user_name
   - Methods:

    - LoginAdminAccount()
    - createCategory()
    - createItem()
    - manageItem()
    - addPromotions()
    - manageOrders()
    - viewUserAccount()
    - viewOrders()
    - viewSupplierAccount()

3. **Member Class**
    - Attributes: user_id, user_password, user_name
    - Methods:
        - LoginMemberAccount()
        - createAccount()
        - giveFeedback()
        - contactSprinkles()
        - viewItem()
        - orderItem()
        - viewUserAccount()
        - viewOrders()
4. **Supplier Class**
    - Attributes: user_id, user_password, user_name
    - Methods:

        - LoginSupplierAccount()
        - viewSupplierAccount()
        - viewSupplierOrder()

5. **Category Class**
    - Attributes: category_id, category_name
    - Methods:
        - addCupcake()
        - removeItem()
        - addItem()
6. **Item Class**
    - Attributes: cup_id, cup_type
7. **Order Class**
    - Attributes: order_id, order_type
    - Methods:
        - addItem()
        - viewPrice()
        - placeOrder()
        - cancelOrder()
8. **Promotion Class**
    - Attributes: promo_id, promo_name
    - Methods: addPromotion()

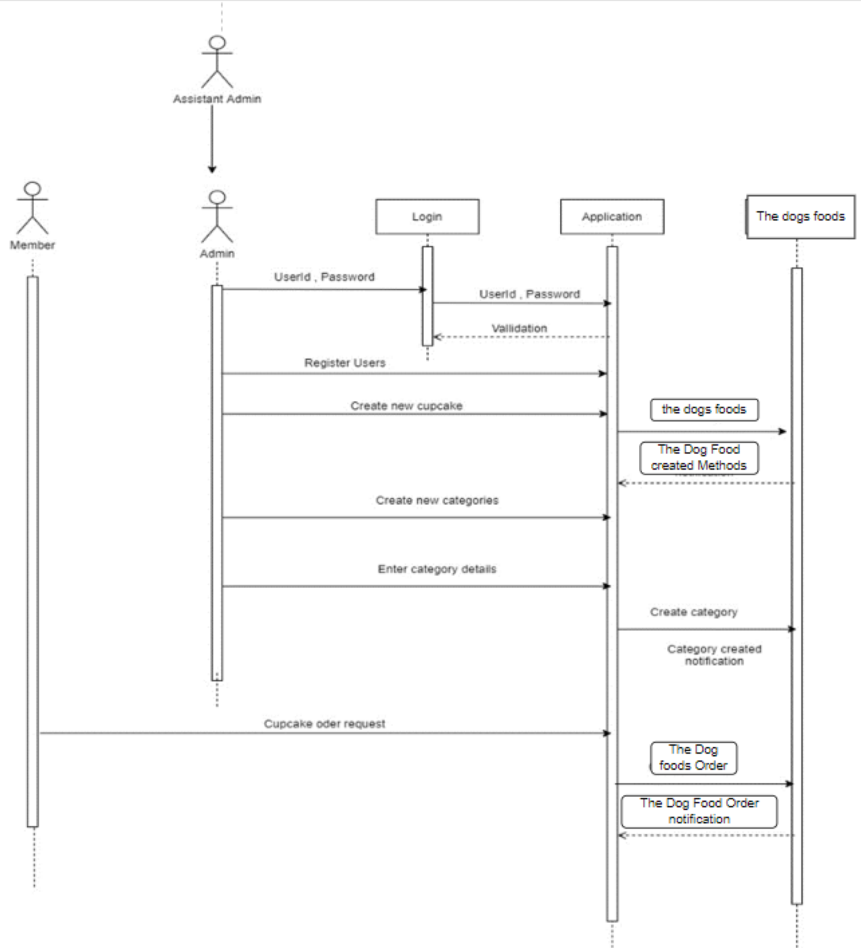
Relationships

- **User** is associated with **Admin**, **Member**, and **Supplier** classes, signifying that these roles are derived from the User.
- **Admin** and **Member** can both place **Order**s.
- **Category** is assigned to **Admin** and **Supplier**.
- **Order** class is associated with **Item** and **Promotion**, where it can add Items and Promotions to an order.

Key Actions:

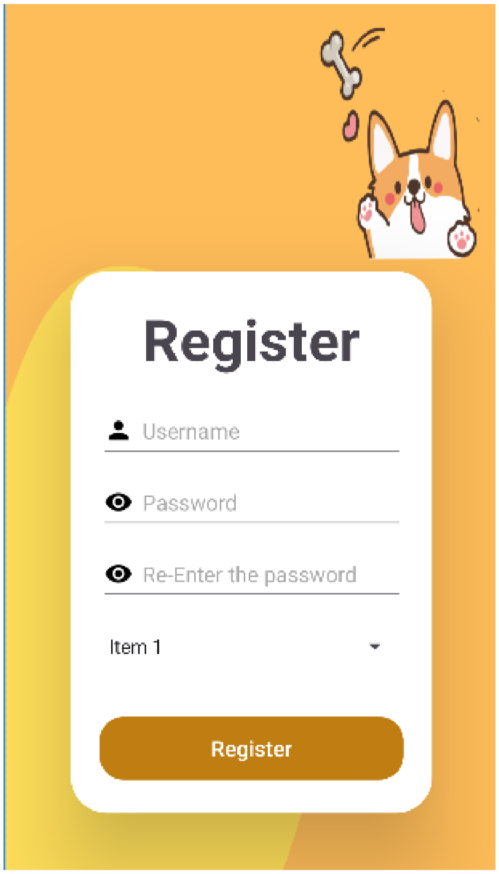
- **Admin** and **Supplier** roles focus on managing categories, items, and orders, as well as creating and viewing accounts.
- **Member** role is more user-oriented, focusing on viewing items, placing orders, and providing feedback.
- **Supplier** can manage their own products and orders, signifying their role in the supply chain.

UML Diagram

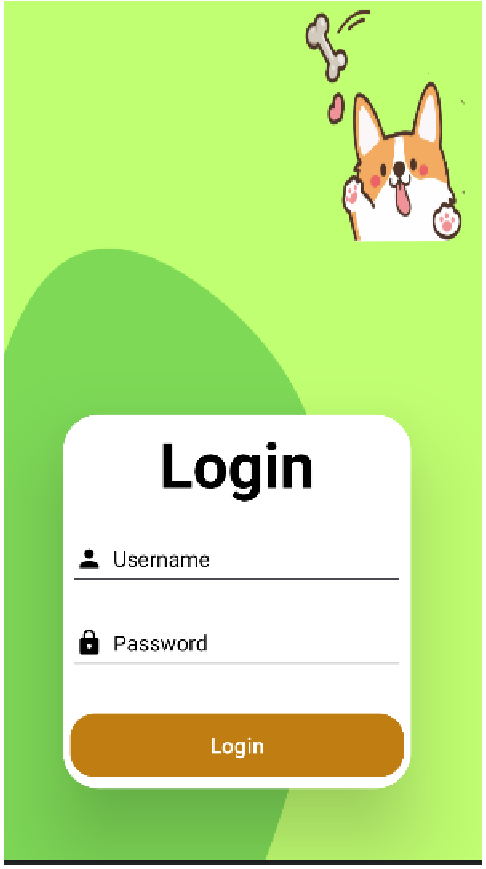

Assistant Admin
Member
Admin
Login
Application
The dogs foods
UserId , Password
UserId , Password
Vallidation
Register Users
Create new cupcake
the dogs foods
The Dog Food created Methods
Create new categories
Enter category details
Create category
Category created notification
Cupcake oder request
The Dog foods Order
The Dog Food Order notification

Task 3

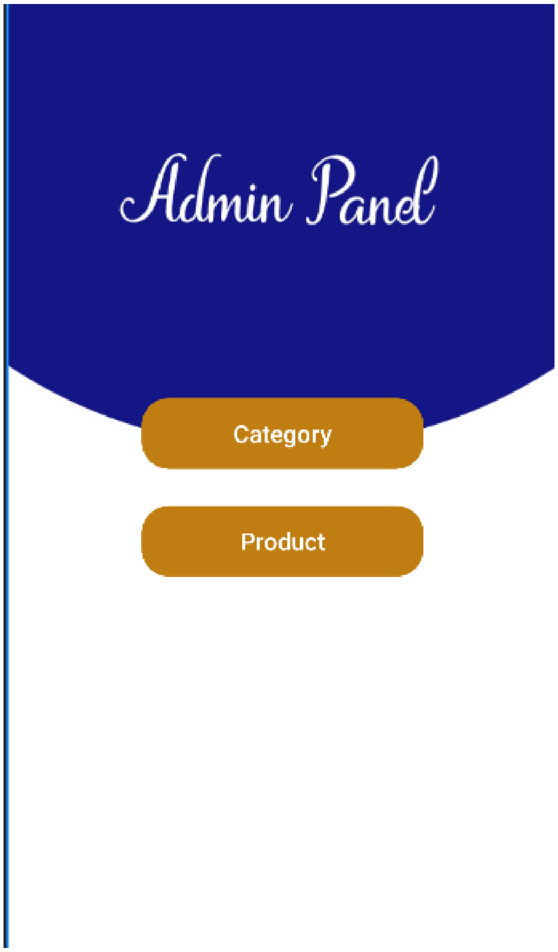
Designs of Interfaces



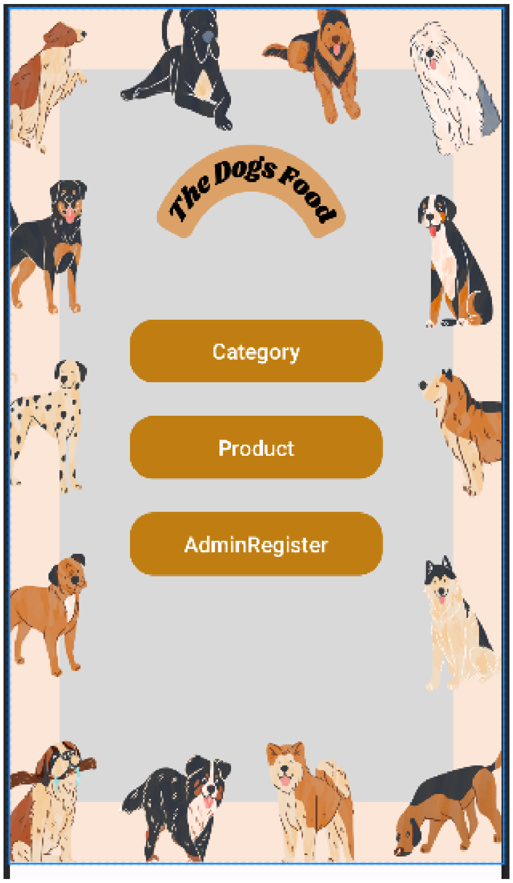
Upon clicking the "Register" button, users are directed to the registration page where they can create new accounts. This page features a spinner tool allowing users to select their role, either as a standard user or an administrator. This intuitive interface simplifies the registration process, ensuring clarity for users and facilitating efficient account creation.



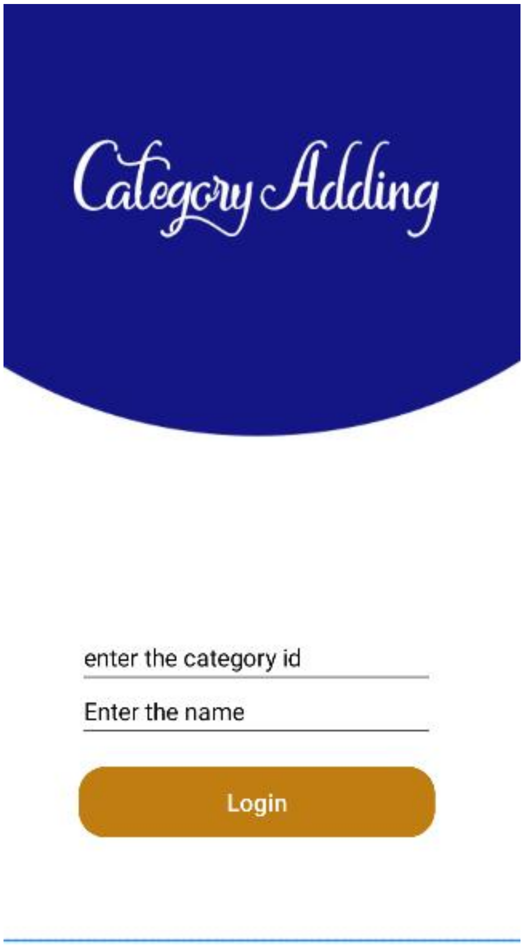
The login page is designed with simplicity, featuring two fields for entering the username and password. These credentials must match the registered pattern established during the respective registration processes for users, administrators, and assistant administrators. This streamlined interface ensures quick access to the system while maintaining security and user-friendliness for efficient login procedures.



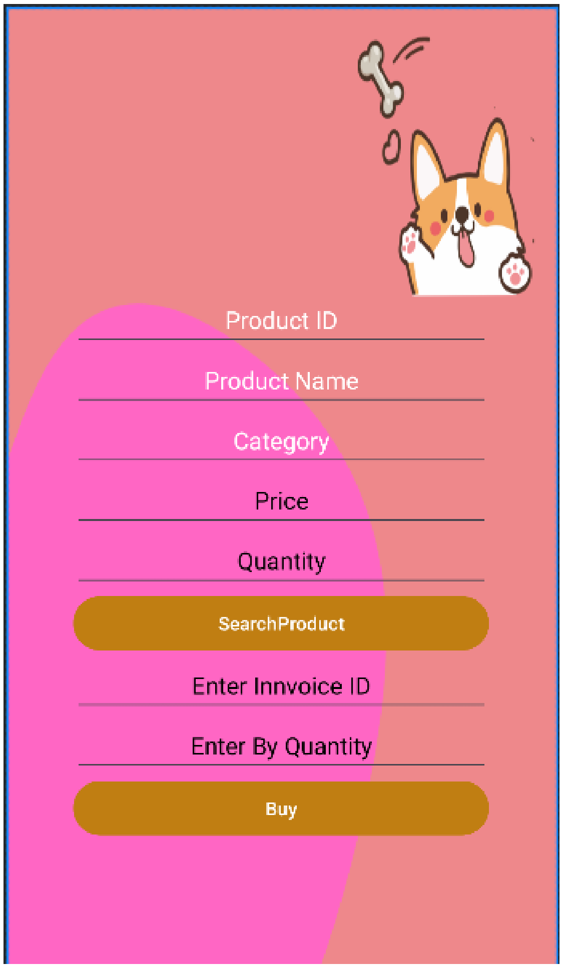
"The main page of the system serves as the primary entry point for users upon accessing the system. It acts as the gateway to the system's functionalities and features, providing users with their initial interaction point. This main interface is meticulously designed to offer a seamless user experience, ensuring users can navigate through the system effortlessly.



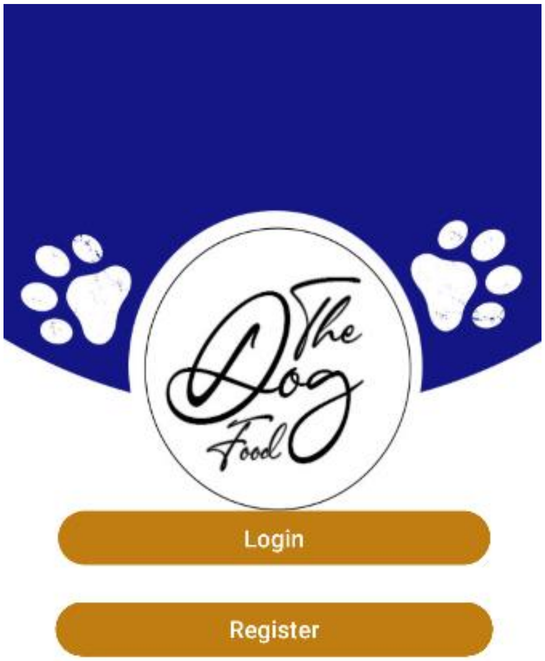
The main page of the admin login provides access to three key functions for administrators: managing categories, adding products to categories, and registering assistant administrators. This streamlined interface empowers administrators to efficiently organize product information, expand the product catalog, and delegate administrative responsibilities as needed.



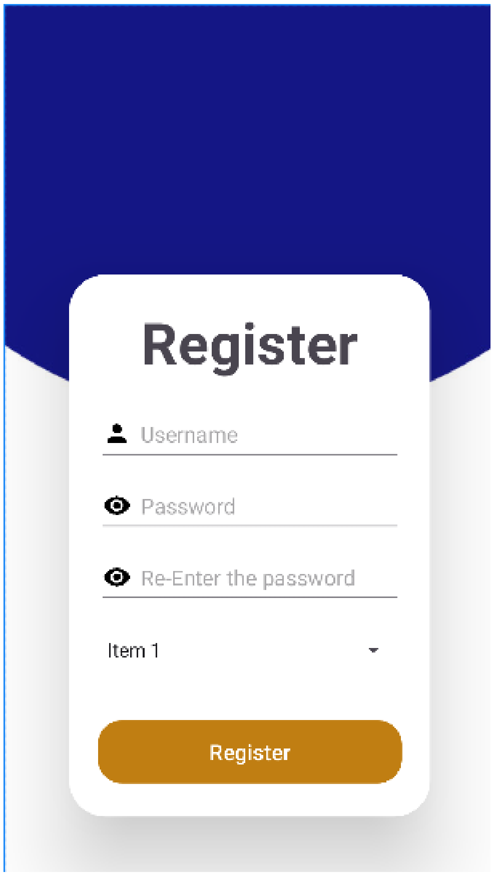
The main interface of the assistant admin grants them the ability to insert both categories and products, akin to the main admin's capabilities. This ensures streamlined administrative operations and facilitates collaborative management of the system.



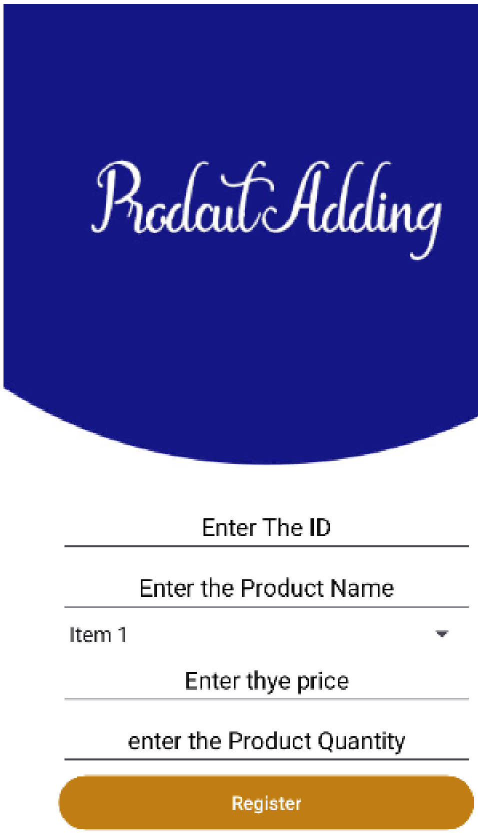
The invoice page features two key functions: searching for product details and buying products. Users can enter a specific product ID to access detailed information about a particular product. Additionally, they have the option to purchase products directly from the invoice page, streamlining the ordering process for enhanced convenience and efficiency.



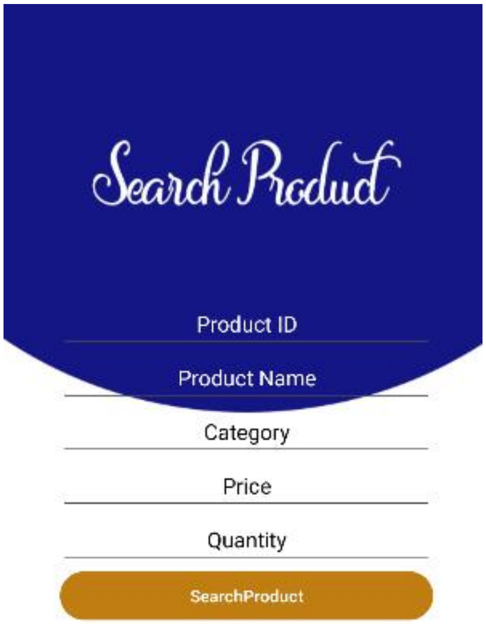
"The main page of the system serves as the primary entry point for users upon accessing the system. It acts as the gateway to the system's functionalities and features, providing users with their initial interaction point. This main interface is meticulously designed to offer a seamless user experience, ensuring users can navigate through the system effortlessly.



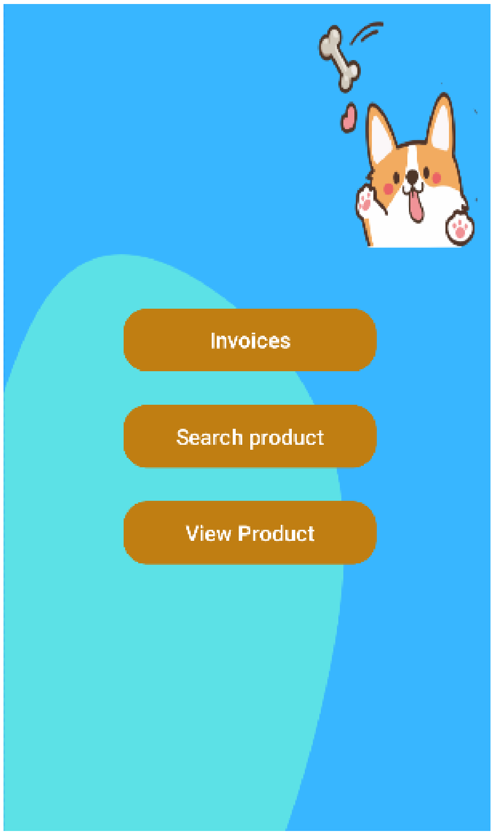
On the product page, administrators can seamlessly add products by selecting the corresponding category in which they wish to insert the product. This user-friendly interface empowers administrators to categorize products efficiently, ensuring they are organized logically within the system. By providing administrators with the flexibility to decide the category for each product, the system accommodates diverse product offerings and simplifies navigation for users. This intuitive approach enhances the overall user experience, facilitating streamlined product management and categorization for administrators.



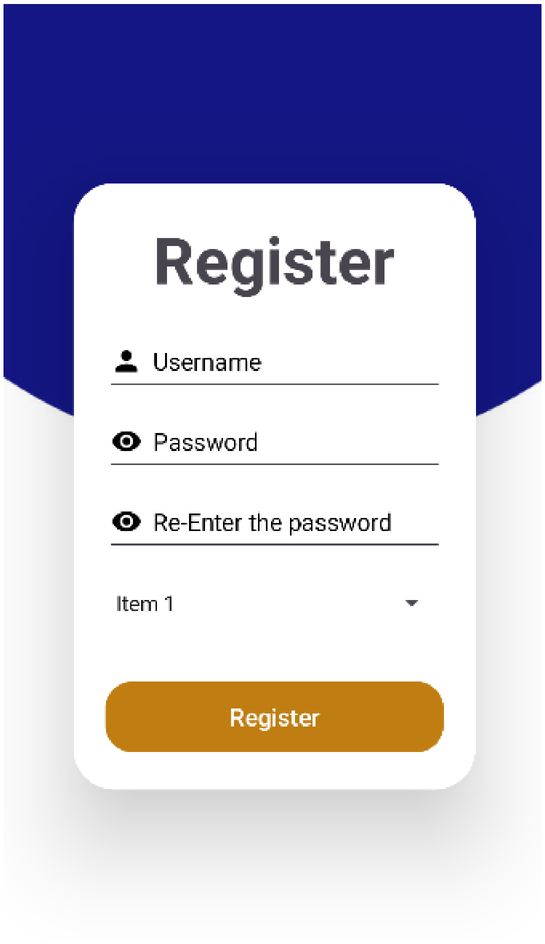
Similarly, in the admin product page, administrators can utilize the search product feature to efficiently access product details. By entering the product ID, administrators can seamlessly retrieve the corresponding product information, streamlining the process of managing products within the system. This intuitive functionality eliminates the need for administrators to navigate through extensive menus or pages, providing immediate access to essential product details. Leveraging this streamlined search capability, administrators can swiftly locate specific products, review their details, and make informed decisions regarding product management and organization.



The invoice page features two key functions: searching for product details and buying products. Users can enter a specific product ID to access detailed information about a particular product. Additionally, they have the option to purchase products directly from the invoice page, streamlining the ordering process for enhanced convenience and efficiency.



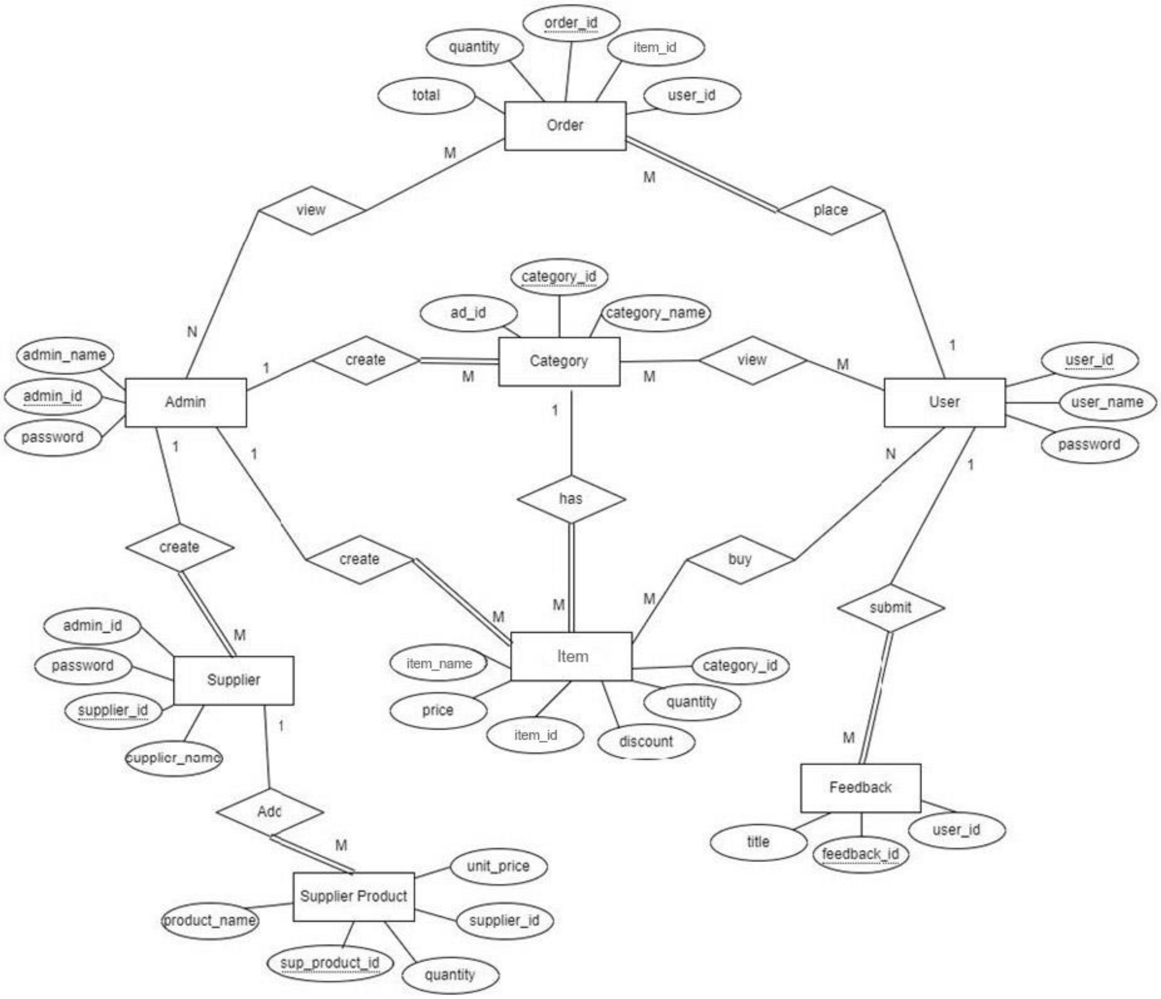
The main page of the admin login provides access to three key functions for administrators: managing categories, adding products to categories, and registering assistant administrators. This streamlined interface empowers administrators to efficiently organize product information, expand the product catalog, and delegate administrative responsibilities as needed.



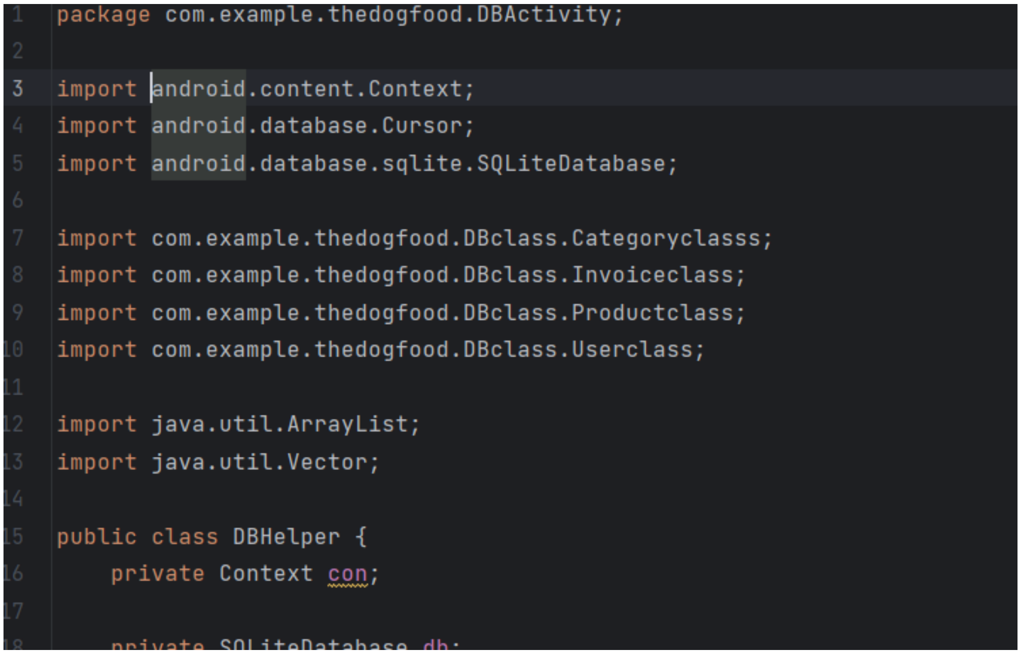
Upon clicking the "Register" button, users are directed to the registration page where they can create new accounts. This page features a spinner tool allowing users to select their role, either as a standard user or an administrator. This intuitive interface simplifies the registration process, ensuring clarity for users and facilitating efficient account creation.

Task 4

ERD Diagram

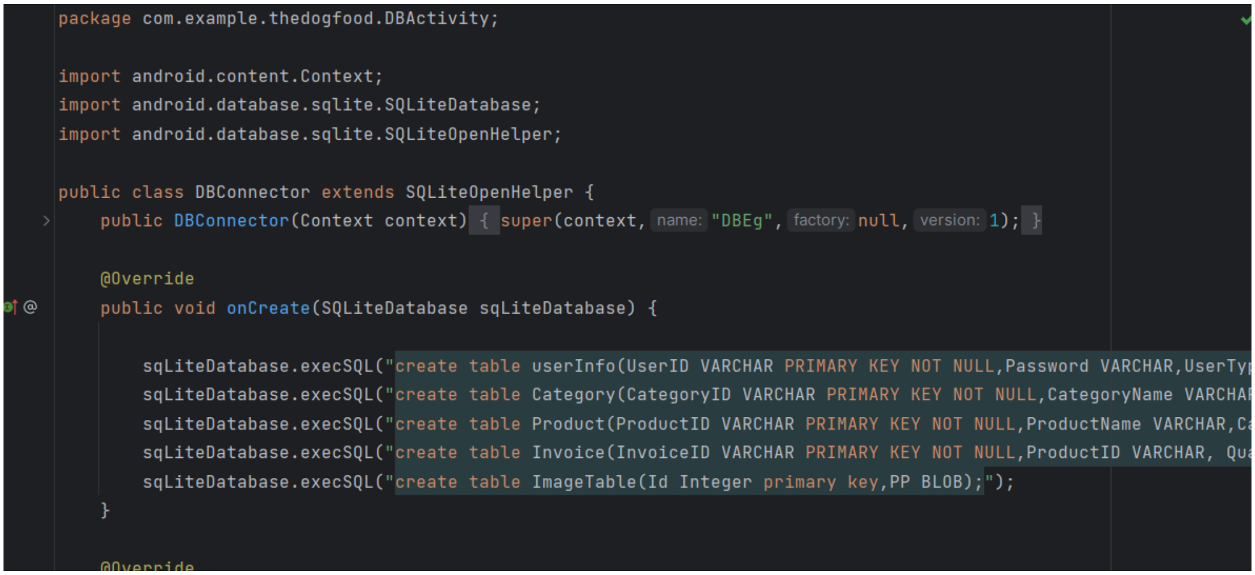


DB Helper



```java
package com.example.thedogfood.DBActivity;

import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;

import com.example.thedogfood.DBclass.Categoryclasss;
import com.example.thedogfood.DBclass.Invoiceclass;
import com.example.thedogfood.DBclass.Productclass;
import com.example.thedogfood.DBclass.Userclass;

import java.util.ArrayList;
import java.util.Vector;

public class DBHelper {
    private Context con;

    private SQLiteDatabase db;
```

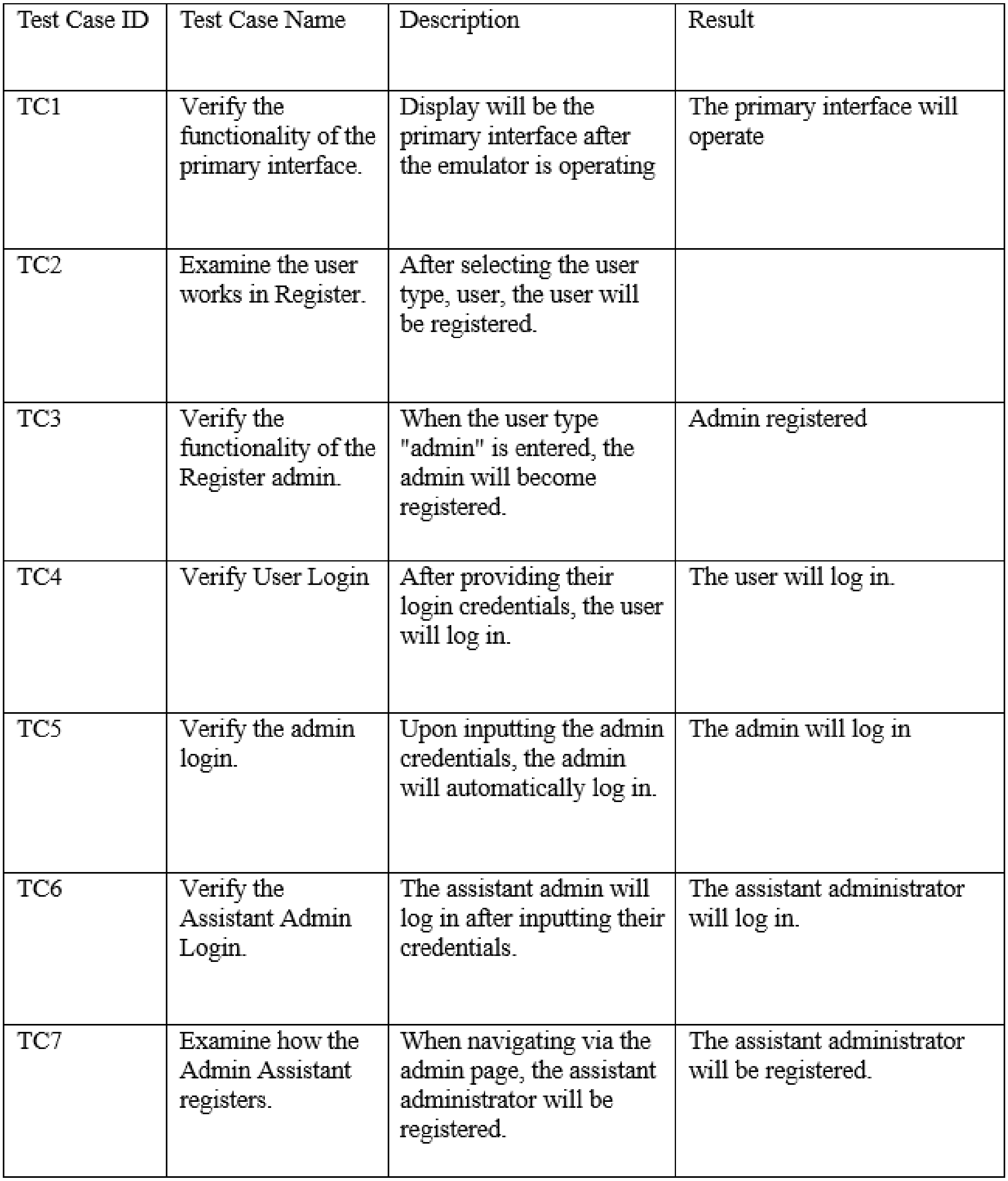
Database Implementation



```java
package com.example.thedogfood.DBActivity;

import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DBConnector extends SQLiteOpenHelper {
    public DBConnector(Context context) { super(context, name: "DBEg", factory: null, version: 1); }

    @Override
    public void onCreate(SQLiteDatabase sqLiteDatabase) {

        sqLiteDatabase.execSQL("create table userInfo(UserID VARCHAR PRIMARY KEY NOT NULL,Password VARCHAR,UserTyp
        sqLiteDatabase.execSQL("create table Category(CategoryID VARCHAR PRIMARY KEY NOT NULL,CategoryName VARCHAR
        sqLiteDatabase.execSQL("create table Product(ProductID VARCHAR PRIMARY KEY NOT NULL,ProductName VARCHAR,Ca
        sqLiteDatabase.execSQL("create table Invoice(InvoiceID VARCHAR PRIMARY KEY NOT NULL,ProductID VARCHAR, Qua
        sqLiteDatabase.execSQL("create table ImageTable(Id Integer primary key,PP BLOB);");
    }

    @Override
```

Task 5

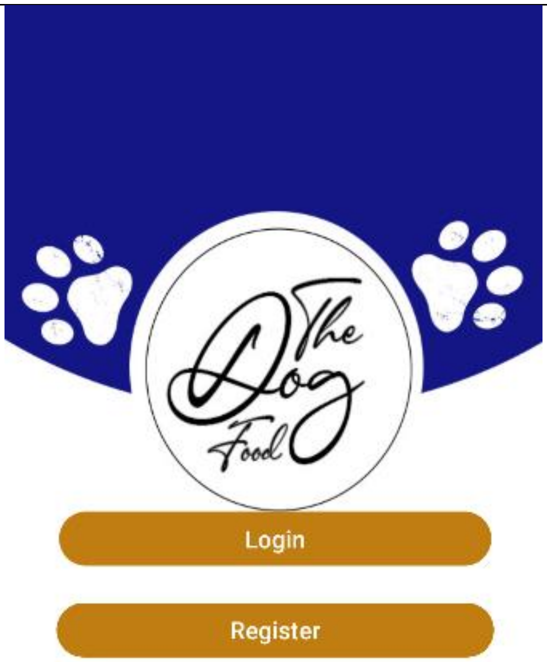
Test Plan

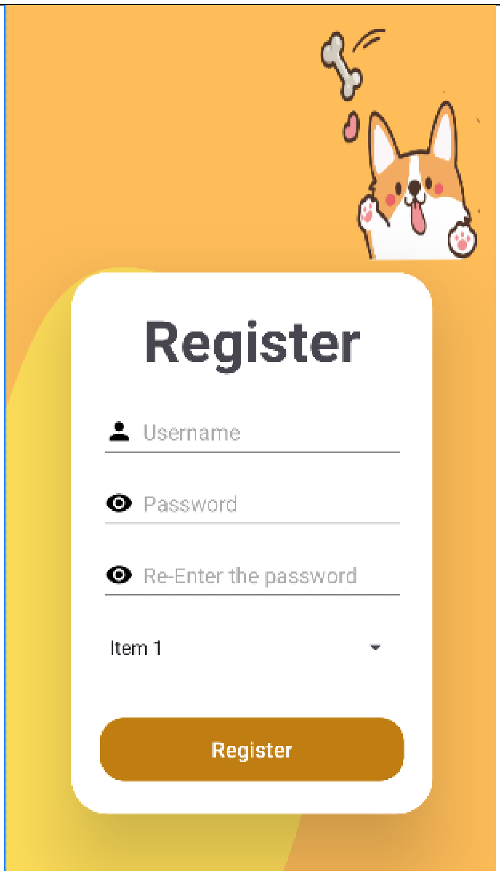


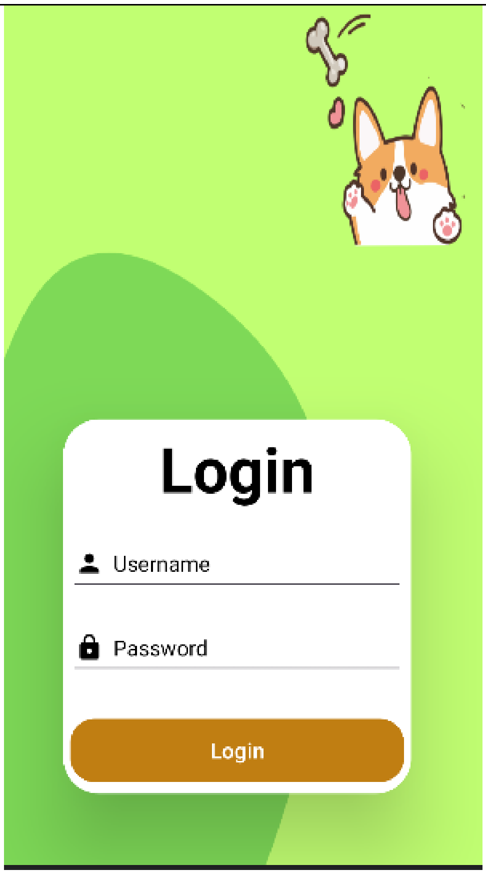
Test Case ID	Test Case Name	Description	Result
TC1	Verify the functionality of the primary interface.	Display will be the primary interface after the emulator is operating	The primary interface will operate
TC2	Examine the user works in Register.	After selecting the user type, user, the user will be registered.	
TC3	Verify the functionality of the Register admin.	When the user type "admin" is entered, the admin will become registered.	Admin registered
TC4	Verify User Login	After providing their login credentials, the user will log in.	The user will log in.
TC5	Verify the admin login.	Upon inputting the admin credentials, the admin will automatically log in.	The admin will log in
TC6	Verify the Assistant Admin Login.	The assistant admin will log in after inputting their credentials.	The assistant administrator will log in.
TC7	Examine how the Admin Assistant registers.	When navigating via the admin page, the assistant administrator will be registered.	The assistant administrator will be registered.

TC8	Check category.	A notice stating that new categories have been added to the system will appear.	The Message will be Displayed.
TC9	Verify the product	product will be added in the system with the product details	product added message will be displayed.
TC10	Verify that the admin assistant's register button functions.	After clicking the admin assistant page, the admin registration page will appear.	The admin registration will show up.
TC11	Verify that the category button functions.	The admin will be sent to the category page after clicking the button.	The category page will appear.
TC12	Verify that the product button functions.	Upon clicking the button, the admin will be sent to the product page.	The product page appears.
TC13	Verify that the "buy product" button functions.	The purchase option will function as soon as the user adds the product.	Purchased item will be shown
TC14	Verify that the product search button functions.	The user will be able to view the product information after entering the product ID.	Details about the product will be shown.
TC15	Examine the "view products" button.	After pressing the button, the bakery's merchandise will be shown.	Details about the product will be shown.

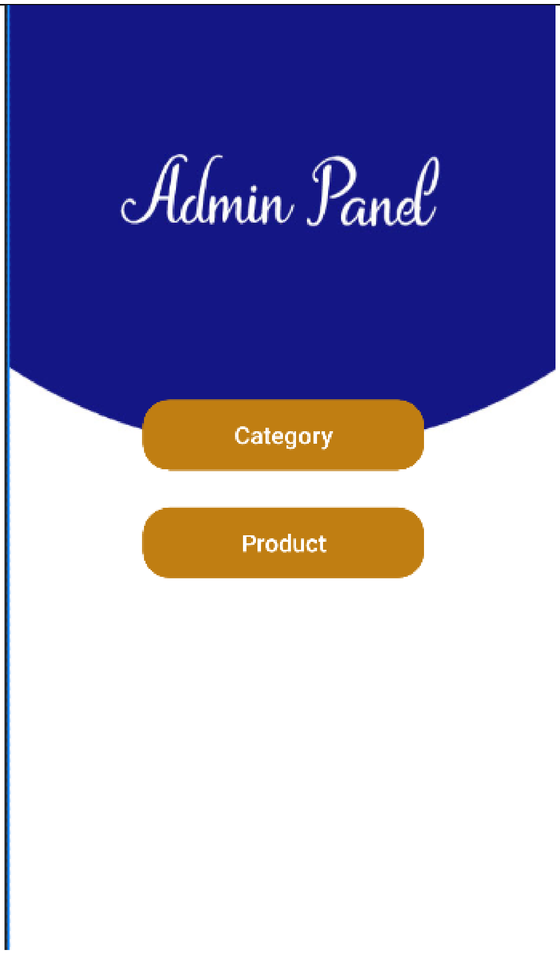
Test Cases

Test Case ID	TC 1
Test Objective	1. Verify if the main interface functions properly.
Test Data	2. Initiate the execution of the main activity.
Expected Result	3. The main interface will commence execution
Actual Result	4. The main interface will start running.
Conclusion	5. The main interface will run.
Screenshot	

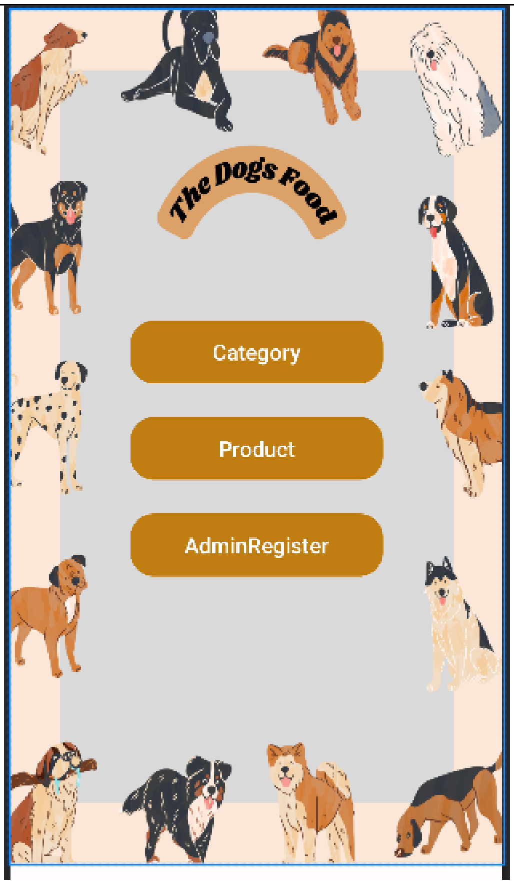
Test Case ID	TC2
Test Objective	Input the user Register into the spinner.
Test Data	Registration of the user will proceed if the user selected in the spinner is entered.
Expected Result	Proceed with user registration upon entering the user selected in the spinner.
Actual Result	User successfully registered.
Conclusion	
Screenshot	 Register Username Password Re-Enter the password Item 1 Register

Test Case ID	TC3
Test Objective	Check the Register Admin.
Test Data	Input the admin into the spinner.
Expected Result	Registration of the admin will proceed if the user selected in the spinner is entered.
Actual Result	Proceed with admin registration upon entering the user selected in the spinner.
Conclusion	Admin successfully registered.
Screenshot	

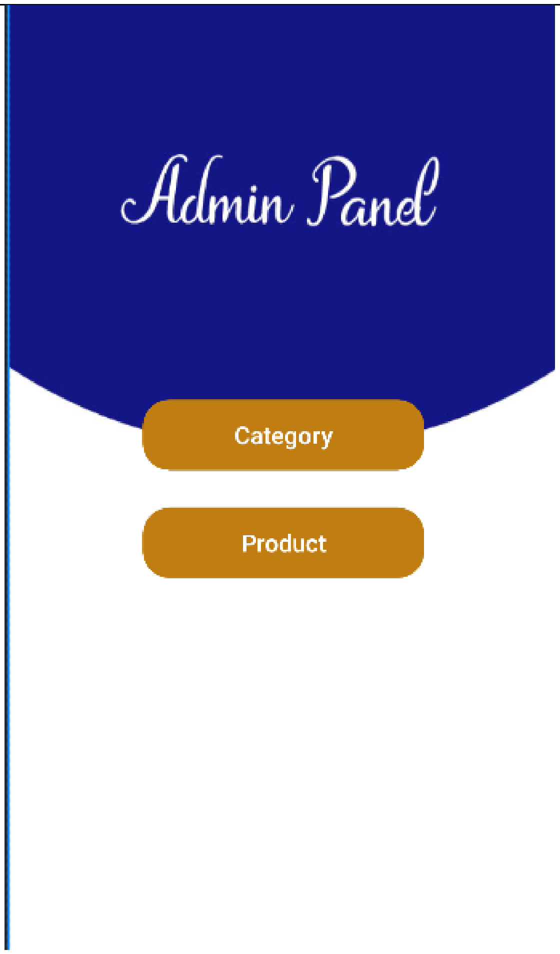
Test Case ID	TC4
Test Objective	Check user Login
Test Data	Enter the login credentials for the user.
Expected Result	User login will be executed if the user credentials were successfully added.
Actual Result	Proceed with user login upon successful addition of user credentials.
Conclusion	User login successful.

Screenshot	

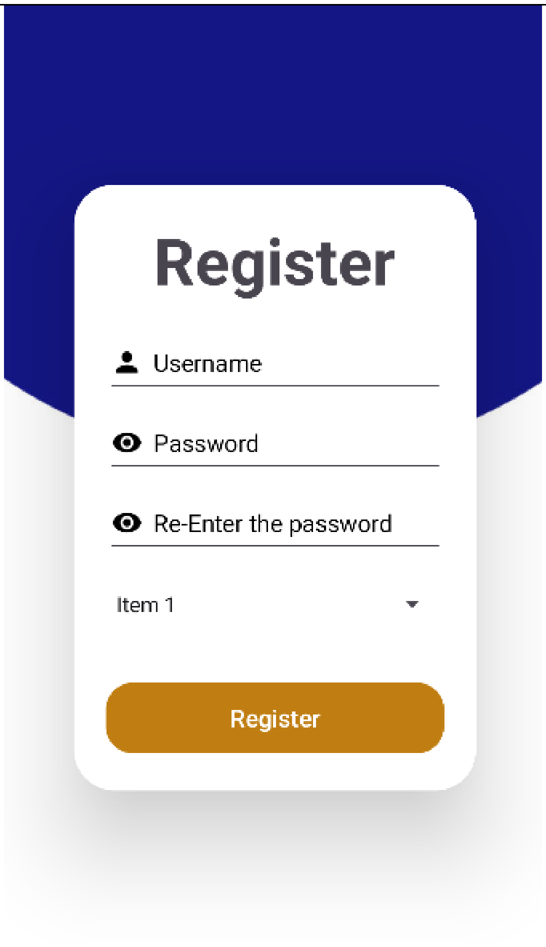
Test Case ID	TC5
Test Objective	Check admin login
Test Data	Enter the login credentials for the admin.
Expected Result	Admin login will be executed if the admin credentials were successfully added.

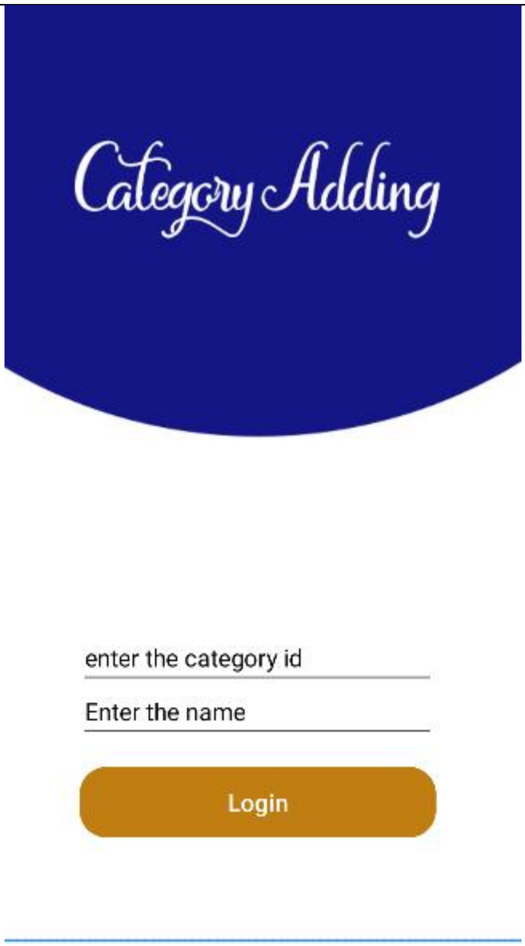
Actual Result	Proceed with admin login upon successful addition of admin credentials.
Conclusion	Admin login successful.
Screenshot	

Test Case ID	TC6
Test Objective	Check Assistant Admin Login
Test Data	Enter the login credentials for the assistant admin login

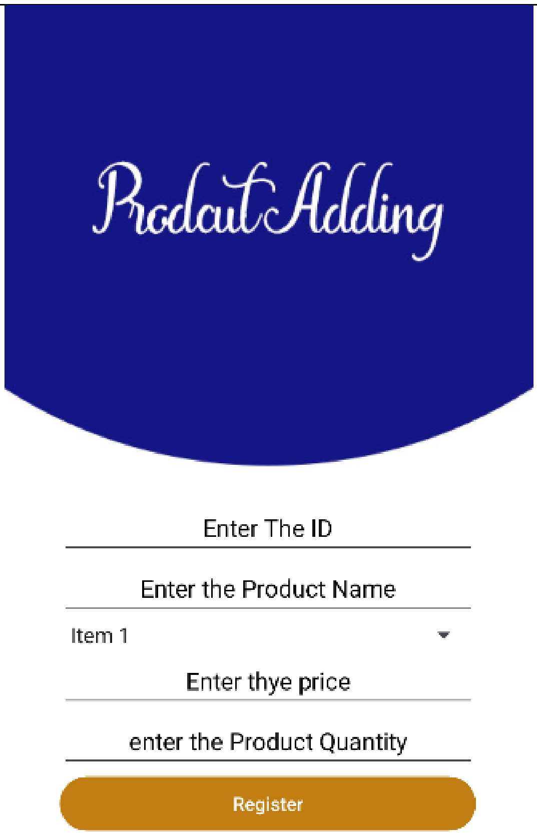
Expected Result	Assistant Admin login will be execute if the assistant admin credentials were added
Actual Result	Assistant Admin login will be execute if the assistant admin credentials were added
Conclusion	Assistant Admin will be login
Screenshot	

Test Case ID	TC7

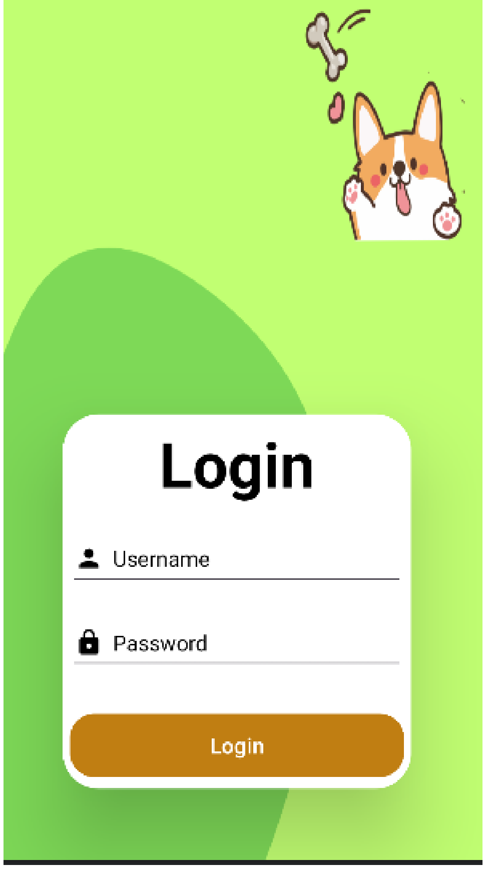
Test Objective	Check Register admin
Test Data	Verify Assistant Admin Registration Functionality
Expected Result	Enter the necessary details to register the assistant admin.
Actual Result	Proceed to register the assistant admin with the provided values.
Conclusion	Assistant admin registration successfully completed.
Screenshot	 Register Username Password Re-Enter the password Item 1 Register

Test Case ID	TC8
Test Objective	Check Categories
Test Data	Enter the necessary details in the category interface.
Expected Result	Fill in the details for the categories.
Actual Result	Fill in the details for the categories.
Conclusion	Confirmation message will be displayed upon successful addition of categories.
Screenshot	 Category Adding enter the category id Enter the name Login

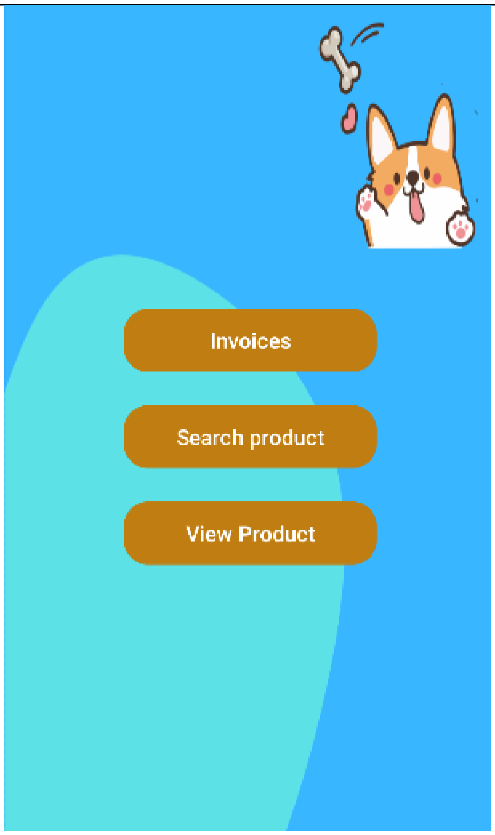
Test Case ID	TC9
Test Objective	Check Product
Test Data	Verify Product Addition
Expected Result	Enter the necessary details in the product interface.
Actual Result	Fill in the product details.
Conclusion	A confirmation message will be displayed upon successful addition of the product.

Screenshot	

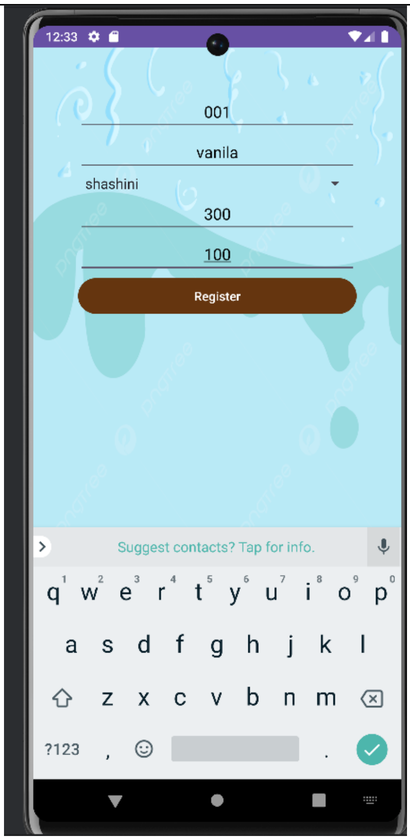
Test Case ID	TC10
Test Objective	Verify Admin Assistant Registration Button Functionality
Test Data	Click on the admin assistant register button.
Expected Result	Upon clicking the register button, the assistant admin page will be displayed.

Actual Result	Ensure that clicking the register button redirects to the assistant admin page.
Conclusion	The admin register page should be displayed upon clicking the register button.
Screenshot	

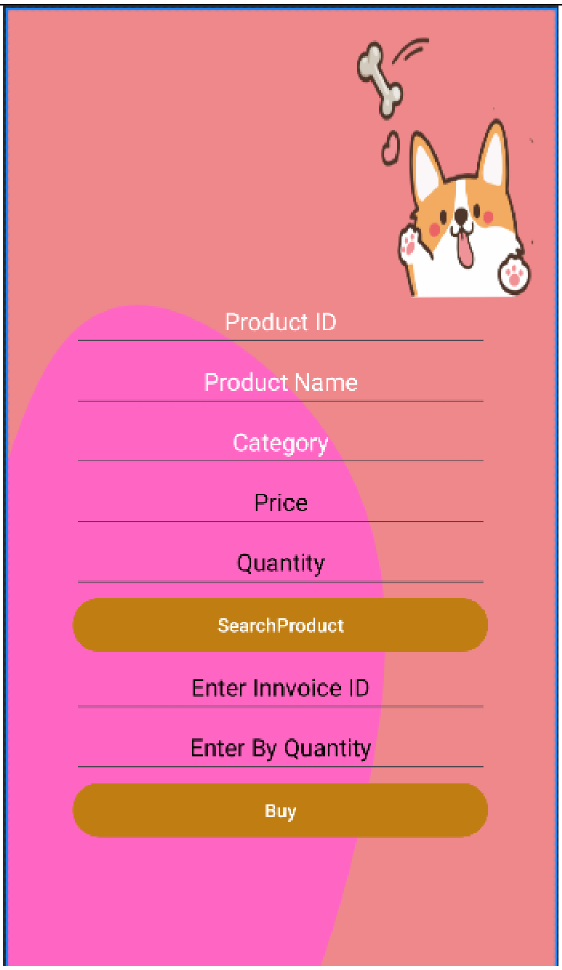
Test Case ID	TC11
Test Objective	Verify Category Button Functionality

Test Data	Click on the category button.
Expected Result	Ensure that clicking the category button redirects to the category page.
Actual Result	The category page should be displayed upon clicking the category button.
Conclusion	The category page should be displayed upon clicking the category button.
Screenshot	

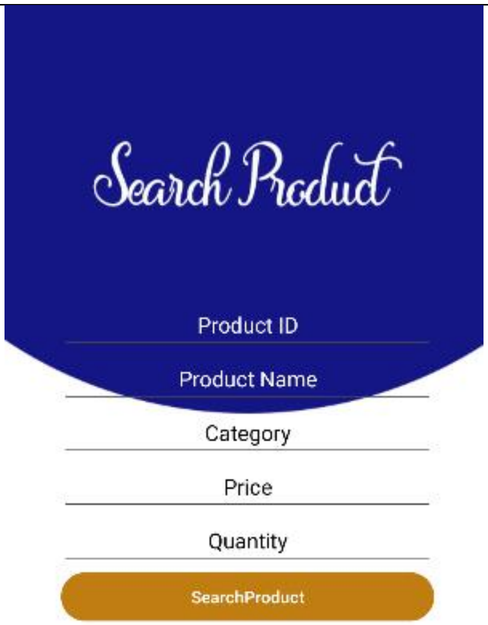
Test Case ID	TC12
Test Objective	Verify Product Button Functionality

Test Data	Click on the product button.
Expected Result	Ensure that clicking the product button redirects to the product page.
Actual Result	The product page should be displayed upon clicking the product button.
Conclusion	The product page should be displayed upon clicking the product button.
Screenshot	 12:33 001 vanila shashini 300 100 Register Suggest contacts? Tap for info. q w e r t y u i o p a s d f g h j k l z x c v b n m ?123

Test Case ID	TC13
Test Objective	Verify Buy Product Button Function.

Test Data	Enter all the necessary details in the buy product page.
Expected Result	Upon entering all the details, the buy product message will be displayed.
Actual Result	Ensure that clicking the buy product button and providing the required details triggers the display of the buy product message.
Conclusion	Ensure that clicking the buy product button and providing the required details triggers the display of the buy product message.
Screenshot	 Product ID Product Name Category Price Quantity SearchProduct Enter Innvoice ID Enter By Quantity Buy

Test Case ID	TC14

Test Objective	Verify Search Product Button Functionality
Test Data	Enter the product ID.
Expected Result	Ensure that upon inserting the product ID, the product details are displayed.
Actual Result	The product details should be displayed upon inserting the product ID and clicking the search product button.
Conclusion	The product details should be displayed upon inserting the product ID and clicking the search product button.
Screenshot	 Search Product Product ID Product Name Category Price Quantity SearchProduct

References

Lucidchart (2019). UML Use Case Diagram Tutorial. [online] Lucidchart.com. Available at: https://www.lucidchart.com/pages/uml-use-case-diagram.

Steele, C. (2023). What is a Mobile Operating System (Mobile OS)? [online] SearchMobileComputing. Available at: https://www.techtarget.com/searchmobilecomputing/definition/mobile-operating-system.

Visual Paradigm (2019). What is Activity Diagram? [online] Visual-paradigm.com. Available at: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram/.

Thak you !!!!