



REPUBLIC OF YEMEN
UNIVERSITY OF ADEN
FACULTY OF ENGINEERING
DEPT. OF INFORMATION TECHNOLOGY



Shopping Without Limits App

E-commerce Mobile Application

Presented By

Abdulbari Sultan Ahmed 217007

Tareq Ahmed Saeed 217017

Supervisor:

Dr. Shada Mabger

2024-2025

Acknowledgment

We feel great pleasure that we present the report on our project work at the end of eighth semester. We take this opportunity to share a few words of gratitude to all those who have supported us in making it possible. We extend our heartfelt gratitude to our project guide **Dr. Shada Mabger** for her continuous guidance and approachability. We would like to express our gratitude towards her constant encouragement, support and guidance throughout the development of the project. Our regular meetings proved to be a boon in the timely completion of this stage of the project.

Abstract

Mobile applications and e-commerce are tools for accessing the Internet and purchasing products and services. These applications are constantly evolving due to the high rate of technological advancement being made. The intention of this thesis project was to develop a user-friendly mobile e-commerce application that allows users to make their life easier by purchasing online. The app was made utilizing Android Studio and the flutter framework.

The application architecture was built using the MVC paradigm, which incorporates all the well-known provider patterns. The databases were built with **Supabase** and were developed to authenticate current users or allow customers to generate an id by signing up. The free source platforms were used to gather all the product information. Further development ideas were also examined.

The project launched with only a rudimentary grasp of the dart language. Many obstacles and disappointments arose along the road, all of which served as excellent learning opportunities. A better appreciation of how to structure a redevelopment, how to better organize and manage the codebase, and what incremental benefits can be gained by utilizing the flutter framework all seem to be vital insights.

Table of Contents

Acknowledgment.....	I
Abstract.....	II
List of Figures.....	IV
List of Tables	V
List of Abbreviations	V
1. Chapter 1 Introduction	7
1.1 Introduction.....	8
1.1.1 Overview.....	8
1.1.2 Problem Statements.....	8
1.1.3 Project Objectives.....	9
1.1.4 Project Scope	9
2. Chapter 2 Methodology.....	10
2.1 Methodology	11
2.1.1 Flutter	11
2.1.2 Dart Language.....	11
2.1.3 Android Studio.....	11
2.1.4 Supabase.....	12
3. Chapter 3 System Analysis and Requirements.....	13
3.1 System Analysis and Requirements.....	14
3.1.1 Database Design Analysis	14
3.1.2 Use Case Diagram.....	15
3.1.3 Activity Diagram	16
3.1.4 Functional Requirements	18
3.1.5 Non-Functional Requirements	18
3.2 Specific Requirements	18
4. Chapter 4 Design and Implementation.....	20
4.1 Design and Implementation	21
4.1.1 GUI (Graphical User Interfaces).....	21
4.2 Implementation.....	31
4.2.1 Planning	31
4.2.2 MVC (Model-View-Controller).....	31
4.3 Testing	32
4.3.1 Test Cases	32
4.3.2 Types of Testing used	33

Conclusion	35
References	35

List of Figures

Figure 1 Tables Schema Database	14
Figure 2 Use Case Diagram	15
Figure 3 Activity Diagram for Admin Side	16
Figure 4 Activity Diagram for User Side.....	17
Figure 5 Onboarding Screen	21
Figure 6 Login Screen.....	22
Figure 7 Signup Screen.....	22
Figure 8 OTP Screen.....	22
Figure 9 Home Screen	23
Figure 10 Category Screen.....	23
Figure 11 Profile Screen	23
Figure 12 Specific Category Screen	24
Figure 13 Product Screen.....	24
Figure 14 Cart Screen	24
Figure 15 Tables in Supabase	30
Figure 16 Supabase Dashboard	30

List of Tables

Table 1 Specific Requirements	19
-------------------------------------	----

List of Abbreviations

No.	Abbreviation	Expanded form
1	OS	Operating System
2	iOS	iPhone Operating System
3	API	Application Programming Interface
4	SDK	Software Development Kit
5	Json	JavaScript Object Notation

1. Chapter 1 Introduction

1.1 Introduction

1.1.1 Overview

In this thesis, the Flutter technology is utilized to develop an e-commerce mobile application. E-commerce, or electronic commerce, is a technique of purchasing goods and services through the internet that is growing in popularity and effectiveness for online businesses. Ecommerce offers a number of benefits, the most significant of which is that it is inexpensive and convenient.

The "Shopping Without Limits" application is an e-commerce platform built using Flutter and based on the Dart programming language. The project follows the MVC (Model-View-Controller) architectural pattern. The app displays products that are uploaded by wholesalers or retailers through a separate application designed for product uploads. The products have specific details such as their price, description, and storage location, which is identified by the storage name, number, and mapped coordinates.

Consumers can browse through products, view their details, and add them to the shopping cart. After that, they can proceed with the checkout process, where the total product cost and delivery fees are calculated. Delivery charges are based on distance, with a rate of 500 riyals per kilometer. The app offers various payment methods such as cash on delivery, bank deposits, or wire transfers. Additionally, the app tracks the status of orders with phases like "Under Review," "Confirmed," "Cancelled," or "Delivered."

1.1.2 Problem Statements

E-commerce provides an easy way to sell products to a large customer base. However, there is a lot of competition among multiple e-commerce sites. When users land on an e-commerce site, they expect to find what they are looking for quickly and easily. Also, users are not sure about the brands or the actual products they want to purchase. They have a very broad idea about what they want to buy.

1.1.3 Project Objectives

- Improve customer experience by more attractive user interface.
- To assists in the improvement of E-commerce standard, carefully crafted products for distribution within the showcase.
- Using faster server-to-client response, the framework allows a buyer to quickly explore or search for an item.
- To foster customer loyalty.
- To drive sales.
- To Boost customer engagement.
- To Get valuable customer insights.
- To Gain a competitive advantage.

1.1.4 Project Scope

- Enables users to shop from anywhere, anytime, while they travel, during workout, etc.
- The presence of more mobile applications influences the purchase decisions of those who wish to buy products only after comparing the prices, features and other attributes of all other products.
- Collects the customer feedback and data for survey campaigns.
- Keeps a track of the usage pattern and shopping history of customers.
- Facilitates easy shopping of different items, and sharing of various company details without customers being present physically.

2. Chapter 2 Methodology

2.1 Methodology

2.1.1 Flutter

This application is built on Flutter technology. Flutter offers several advantages because it is a Google product. The following is a quick overview of flutter.

The app development process is revolutionized by Flutter. You can design, test, and publish stunning mobile, web, desktop, and embedded apps by writing one-time code. Google built Flutter as an open-source project. It's used to make hybrid apps for Android, iOS, Linux, macOS, Windows, Google Fuchsia, and the web from a single codebase. (Flutter)

2.1.2 Dart Language

Dart is a client-oriented programming language that enables rapid app development across all platforms. Its primary goal is to create one of the most productive languages on a variety of platforms. Both the server and the user will benefit from it. The Dart SDK includes the Dart VM compiler as well as the dart2js tool, which generates the JavaScript version of a Dart Script so that it may be executed on sites that don't support Dart. It is a very similar object-oriented programming language to C++. Dart is a popular programming language for creating single-page websites and online apps. (Dart).

Dart is designed to provide logic and, as a result, a beautiful user interface. For mobile, desktop, and backend apps, compile specialized machine code. Alternatively, for web use, compile to JavaScript.

2.1.3 Android Studio

It's a Google-created integrated development environment (IDE). The goal of Android is to speed up progress and make it easier for users to create high-quality apps for Android devices. Commonly used operating frameworks, such as Mac and Windows, allow variations of this IDE. It also provides a development kit and plugins for developers. cross-platform application support (IDE).

2.1.4 Supabase

It is an open-source backend-as-a-service (BaaS) platform designed to simplify the development of modern web and mobile applications. It provides developers with a scalable and feature-rich backend, similar to Firebase, but built on top of PostgreSQL.

3. Chapter 3 System Analysis and Requirements

3.1 System Analysis and Requirements

In this chapter, it will show how the project is working in diagram and what is the requirements to clarify step by step every single operation in app.

3.1.1 Database Design Analysis

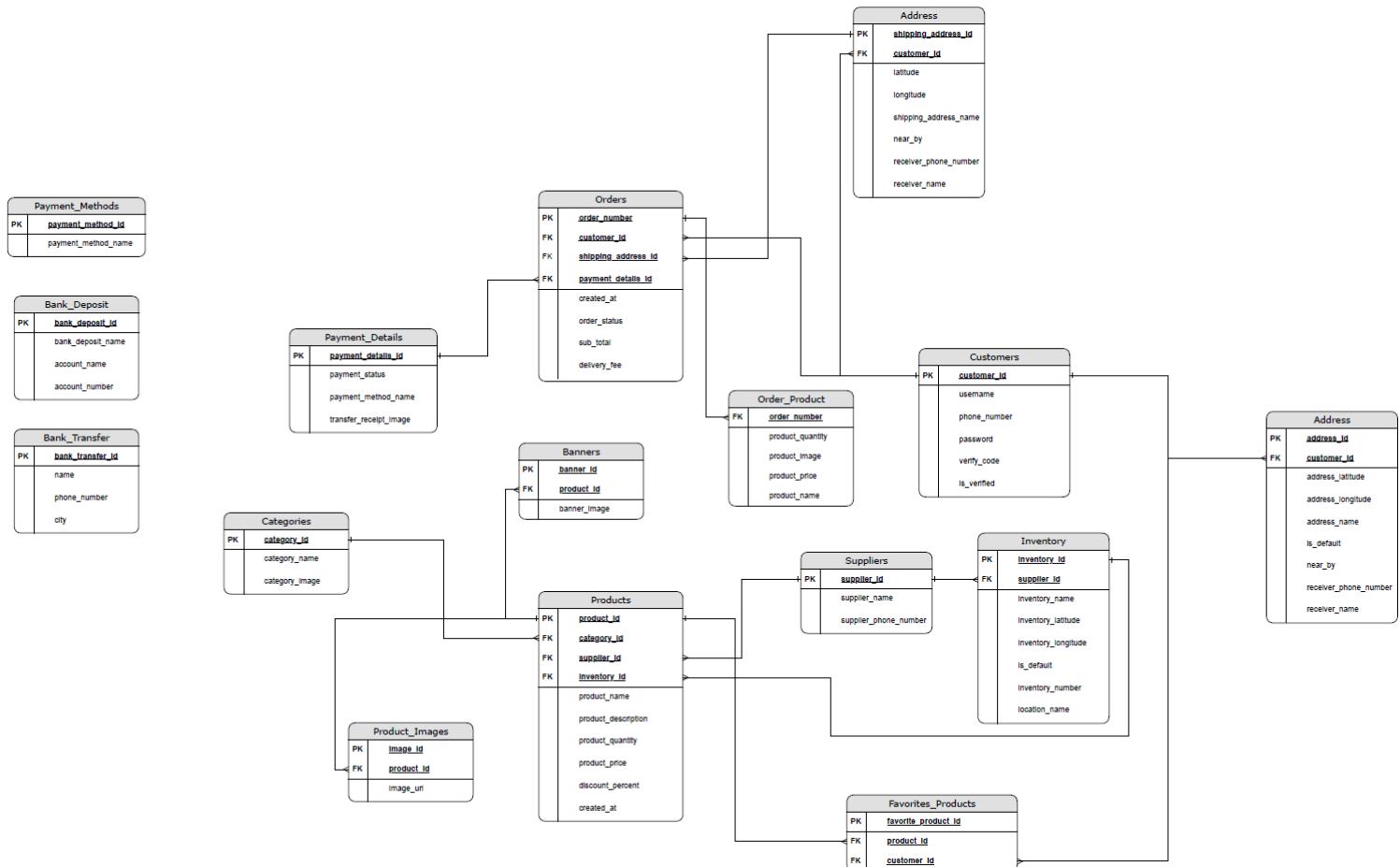


Figure 1 Tables Schema Database

3.1.2 Use Case Diagram

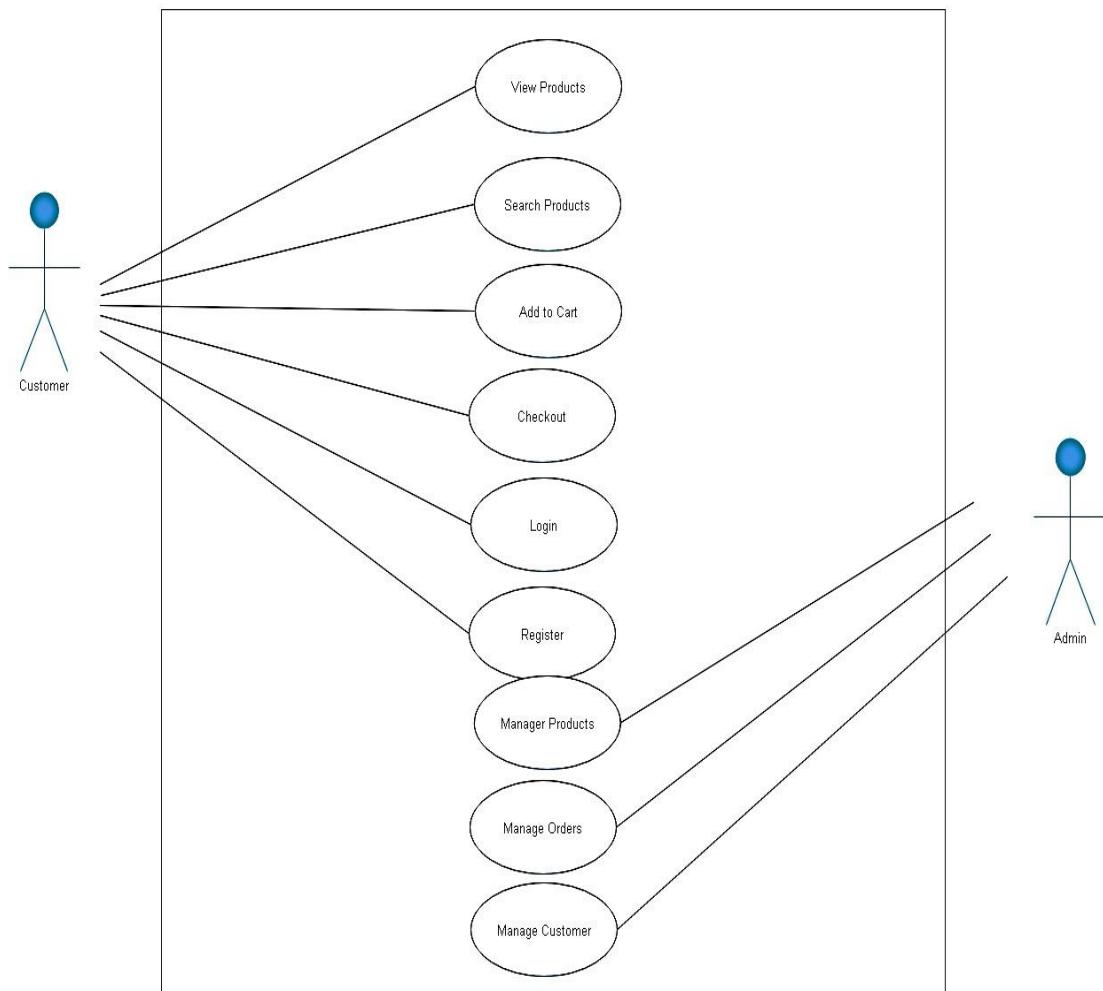


Figure 2 Use Case Diagram

3.1.3 Activity Diagram

- **Activity Diagram for Admin Side**

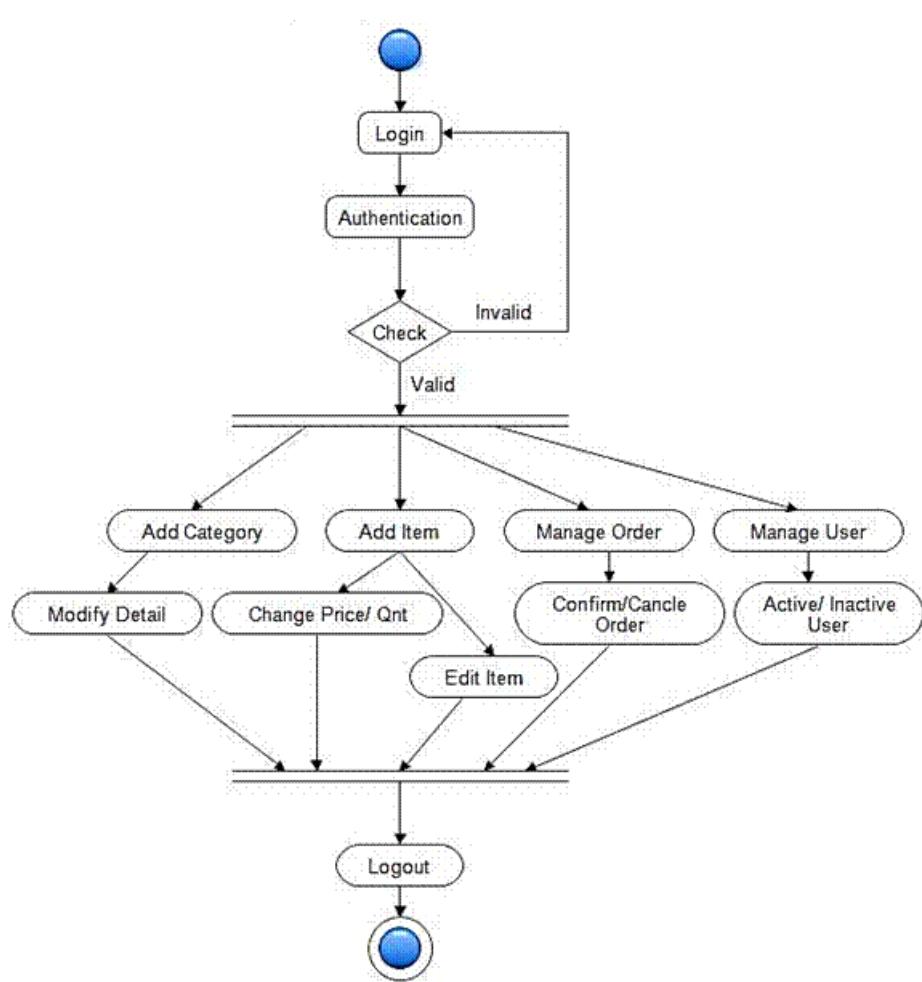


Figure 3 Activity Diagram for Admin Side

- **Activity Diagram for User Side**

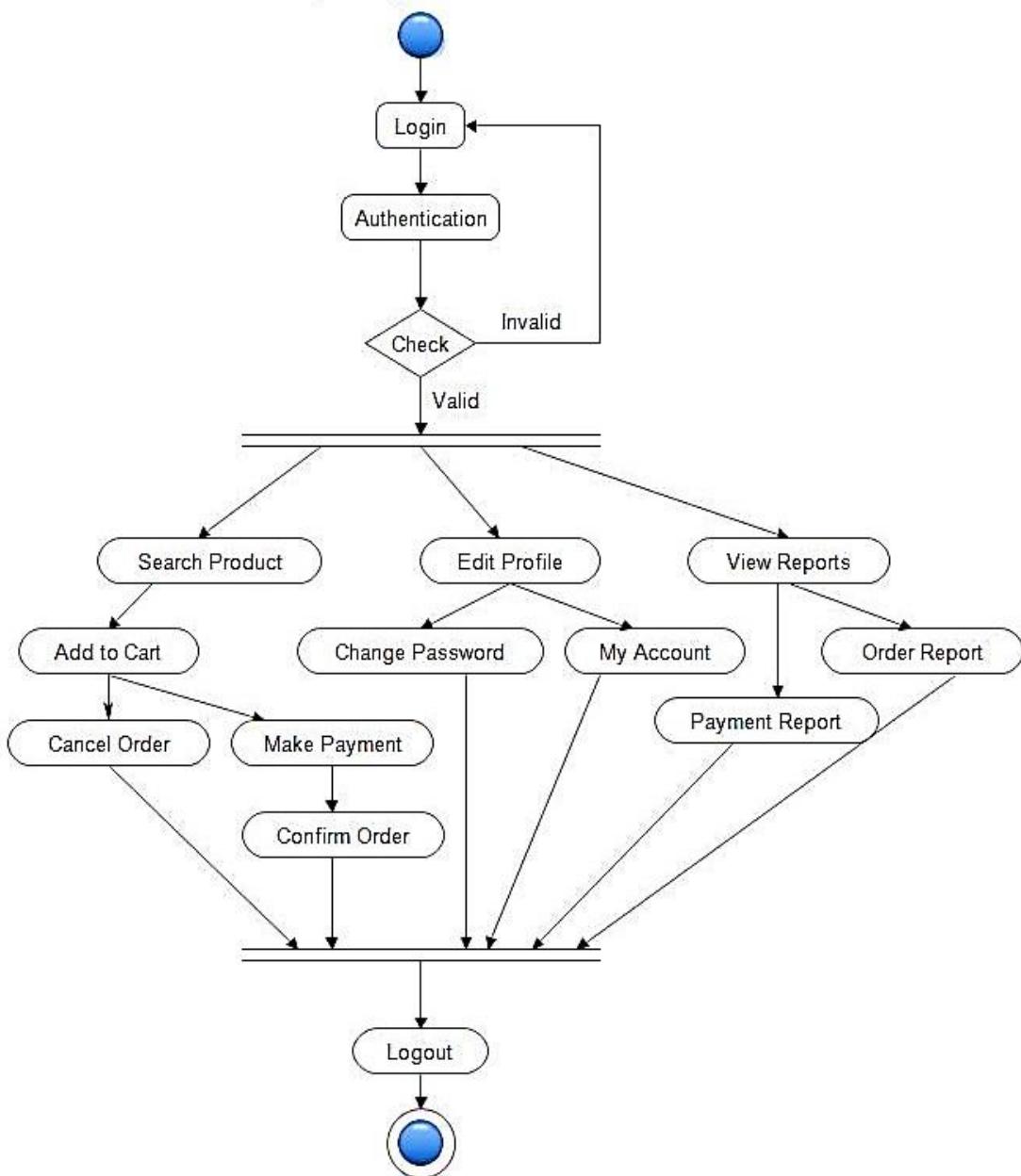


Figure 4 Activity Diagram for User Side

3.1.4 Functional Requirements

The functional requirements for this application that are necessary to complete this project are as follows:

- Application must start when client clicks on app icon
- Application must show the home page of the application first
- Application must provide the selected category's content list for the client
- Client must be able to view the products
- Client must be able to view the products image
- Application must provide the selected product's description for the client
- Client must be able to add the product to the cart
- Application should add the product to the customer's shopping cart.
- Application must provide the customer's information form for the client.
- Client must send the provided credentials to the application.

3.1.5 Non-Functional Requirements

The non-functional requirements for this application that are necessary to complete this project are as follows:

- Application should have an interface to display products to the customer.
- Application should have an interface to display list of categories
- Application should be able to display list of categories, so customers easily click on them
- Application should be able to display a product's description
- Application should have an interface for the customer to add a product to the shopping cart.

3.2 Specific Requirements

Hardware Requirements	Software Requirements

Android device	Android Studio or Visual Studio Code
	Flutter
	Android OS (minimum Android API 21: 5.0 Lollipop)
	Minimum RAM 500 MB.

Table 1 Specific Requirements

4. Chapter 4 Design and Implementation

4.1 Design and Implementation

4.1.1 GUI (Graphical User Interfaces)

4.1.1.1 Customer app

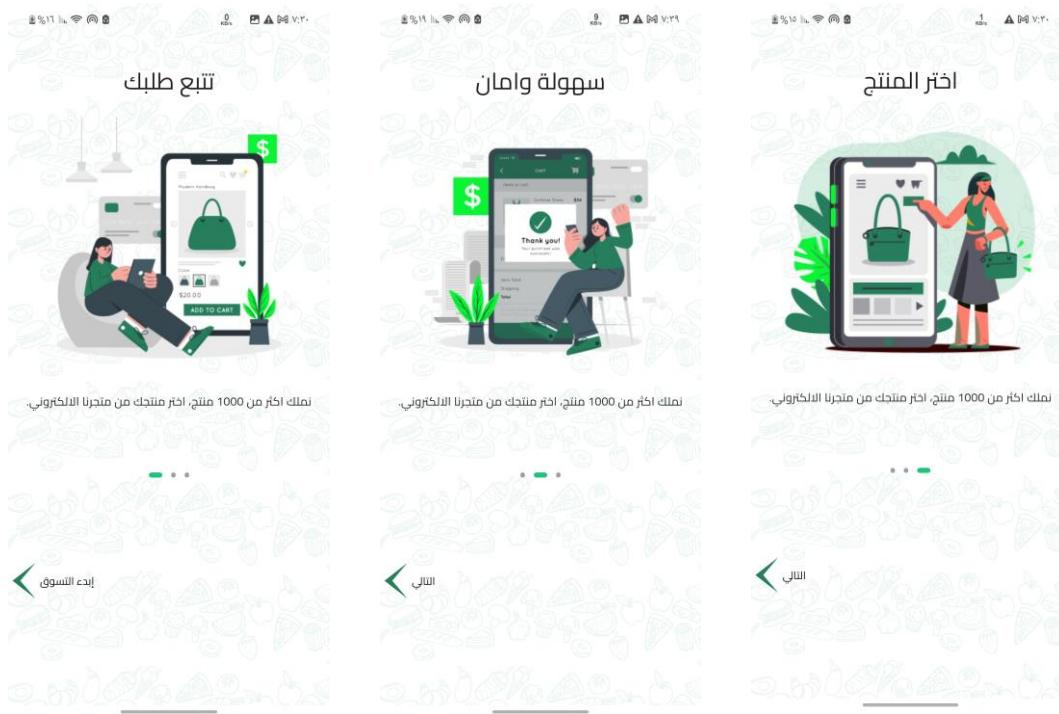


Figure 5 Onboarding Screen



Figure 6 Login Screen



Figure 7 Signup Screen



Figure 8 OTP Screen

ادخل 1 في كل الخانات لانه لم يتم معالجة
هذه الصفحة



Figure 9 Home Screen



Figure 10 Category Screen



Figure 11 Profile Screen



Figure 12 Specific Category Screen



Figure 13 Product Screen

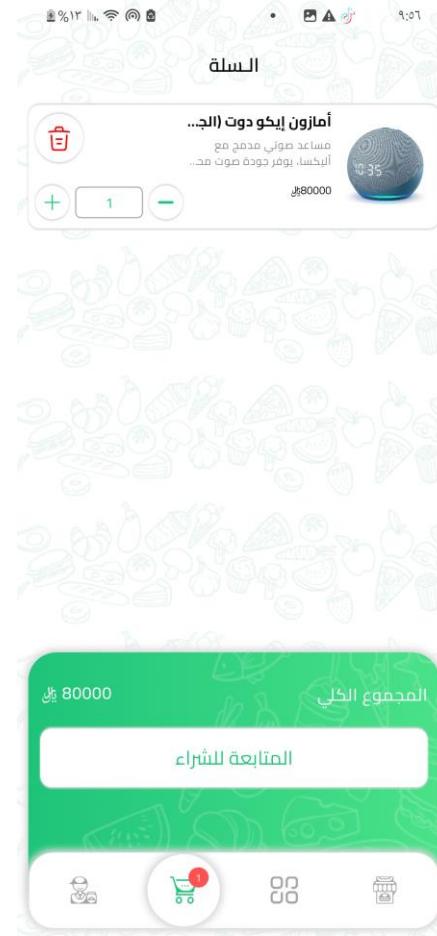


Figure 14 Cart Screen



Figure 4.2.11 Buy Screen

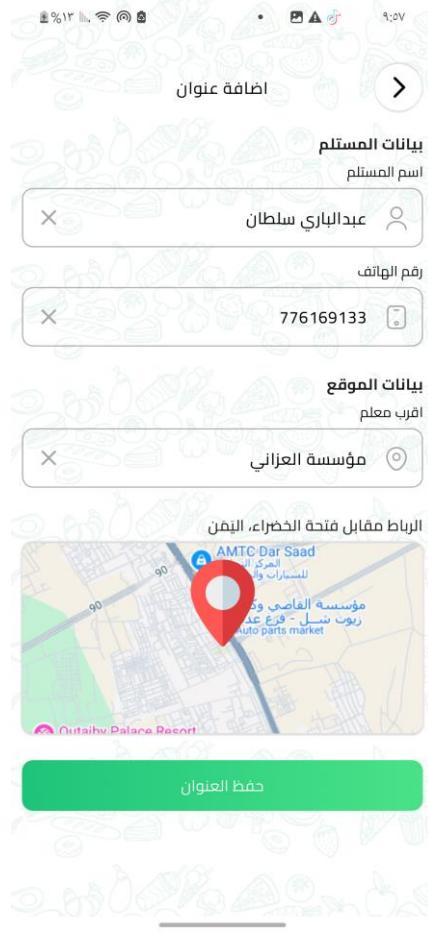


Figure 4.2.12 Add Address Screen



Figure 4.2.13 Map Screen

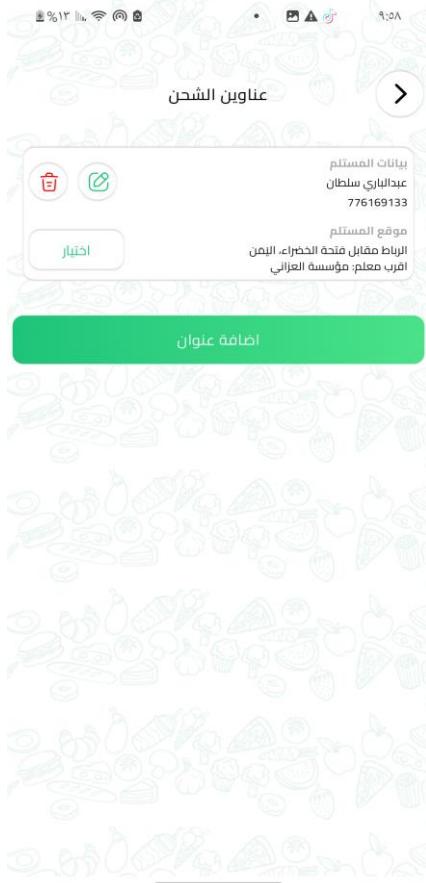


Figure 4.2.14 Address List Screen



Figure 4.2.15 Bank deposit payment method Screen

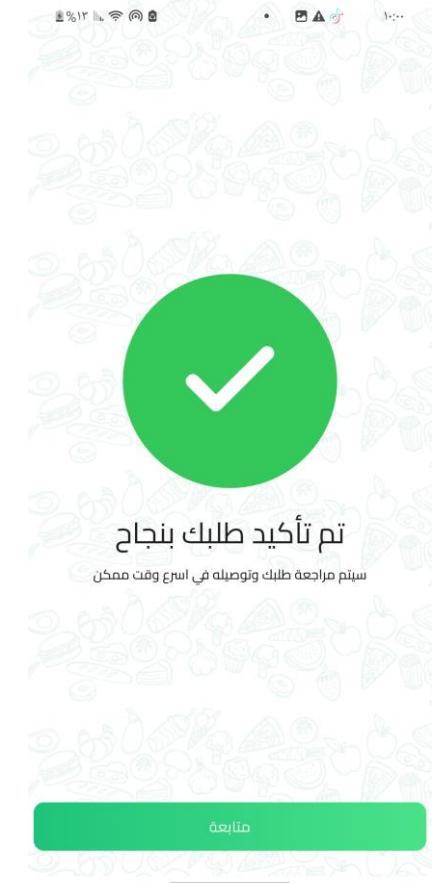


Figure 4.2.16 Successful Complete buy products Screen



Figure 4.2.17 Orders list Screen



Figure 4.2.18 Order details Screen

4.1.1.2 Supplier app



رقم الهاتف

كلمة المرور

تسجيل الدخول



الكمية: 24 سماعات مدبطة	لوحة مفاتيح ميكانيكية (Mec...)
الكمية: 30 مفاتيح ميكانيكية	مسلسلات الحاسوب
الكمية: 20 ماوس ألعاب DPI عالي	كرسي لألعاب
الكمية: 50 دراع تحكم لسلكي	شاشة ألعاب 144 هرتز
الكمية: 5 إضاءة خلفية للمكتبي	بطاقة رسوميات فوود



صور المنتج: (1)

اضافة صور

مستلزمات الحاسوب

اسم المنتج: مفاتيح ميكانيكية

وصف المنتج: لوحة مفاتيح عالية الجودة تعمل بمفاتيح ميكانيكية توفر استجابة سريعة وصوت مميز مناسب للألعاب والكتابة الطويلة.

الكمية: 30 سعر المنتج باليورو: 25000

حذف المنتج

بيانات المدخل

عنوان سبور

رقم المدخل

موقع المدخل: ٧٧٣٩+QHW

العنوان: شيخ عنكان، اليمن شيخ عنكان

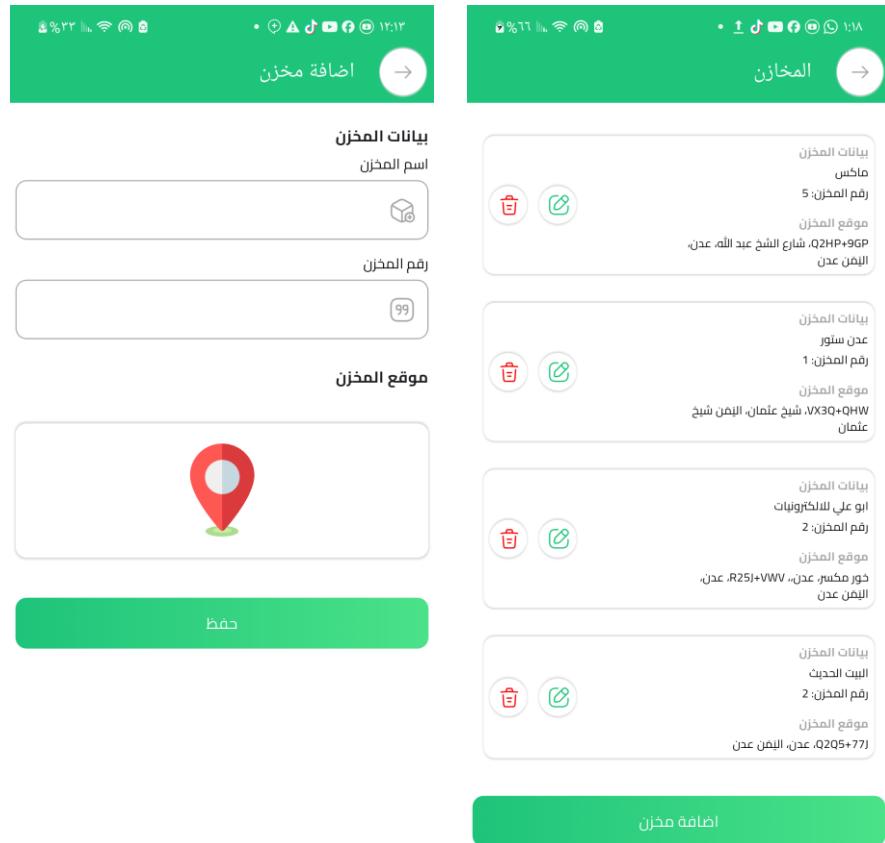
تغير المدخل

اضف منتج

Figure 4.4.1 Login Screen

Figure 4.4.2 Home Screen

Figure 4.4.3 Add product Screen



4.1.1.3 Admin Dashboard

Table Editor									
	prod_id	category_id	supplier_id	inventor_id	product_name	product_description	product_quantity	Unit	Created_at
19	2	1	15	15	لعبة ميكانيكية (Mechanical Keyboard)	جهاز ميكانيكي يحتوي على مفاتيح ملمس ملمس الأصابع والقيمة الطويلة	29	جهاز ميكانيكي يحتوي على مفاتيح ملمس ملمس الأصابع والقيمة الطويلة	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
20	3	1	15	15	ساعات ميكانيكية	ساعات ميكانيكية	22	التي تدور على الميكانيكية	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
21	2	1	15	15	كمبيوتر المحمول	كمبيوتر المحمول	19	كمبيوتر المحمول	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
22	4	1	15	15	كمبيوتر المكتب	كمبيوتر المكتب	22	كمبيوتر المكتب	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
23	4	1	15	15	على مدار الساعة DPI	على مدار الساعة DPI	19	على مدار الساعة DPI	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
24	4	1	16	16	شاشة 144 هرتز	شاشة 144 هرتز	99	شاشة 144 هرتز	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
25	4	1	16	16	ذراع حكم إلكتروني	ذراع حكم إلكتروني	49	ذراع حكم إلكتروني	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
26	4	1	17	17	بطاقة رسومية قوية	بطاقة رسومية قوية	9	بطاقة رسومية قوية	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
27	4	1	15	15	إنسانة حلقية للكتب	إنسانة حلقية للكتب	7	إنسانة حلقية للكتب	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
28	1	1	15	15	ساعة ذكية	ساعة ذكية	29	ساعة ذكية	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
29	1	1	15	15	مساعد منزل ذكي	مساعد منزل ذكي	3	مساعد منزل ذكي	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
30	1	1	15	15	كاميرا مراقبة ذكية	كاميرا مراقبة ذكية	5	كاميرا مراقبة ذكية	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
31	1	1	15	15	منظم حرارة ذكي	منظم حرارة ذكي	9	منظم حرارة ذكي	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
32	1	1	15	15	ميزان ذكي	ميزان ذكي	20	ميزان ذكي	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z
33	1	1	15	15	قفل ذكي	قفل ذكي	4	قفل ذكي	٢٠٢٣-١٠-٢٥T١٤:٣٧:٣٦Z

Figure 15 Tables in Supabase

Figure 16 Supabase Dashboard

4.2 Implementation

4.2.1 Planning

The initial stage in our project's technological execution is to plan what we want to build that will benefit our society. We've attempted to question everything that may be touched by the program up to this point:

- Will the development take place in stages or all at once?
- How long will it take?

Finally, we want to create an ecommerce software that allows users to purchase items without having to go to an actual store or shop. This is an excellent concept for easing the difficulties faced by many folks who are unable to go to the market to purchase goods. An entire store is there at their fingertips; all they have to do is scroll and choose the things that best suit their needs.

4.2.2 MVC (Model-View-Controller)

The app's current design is built on the MVC pattern. It is an application design model comprised of three interconnected parts. They include the model (data), the view (user interface), and the controller (processes that handle input).

The MVC model or "pattern" is commonly used for developing modern user interfaces. It provides the fundamental pieces for designing a program for desktop or mobile, as well as web applications. It works well with object-oriented programming, since the different models, views, and controllers can be treated as objects and reused within an application.

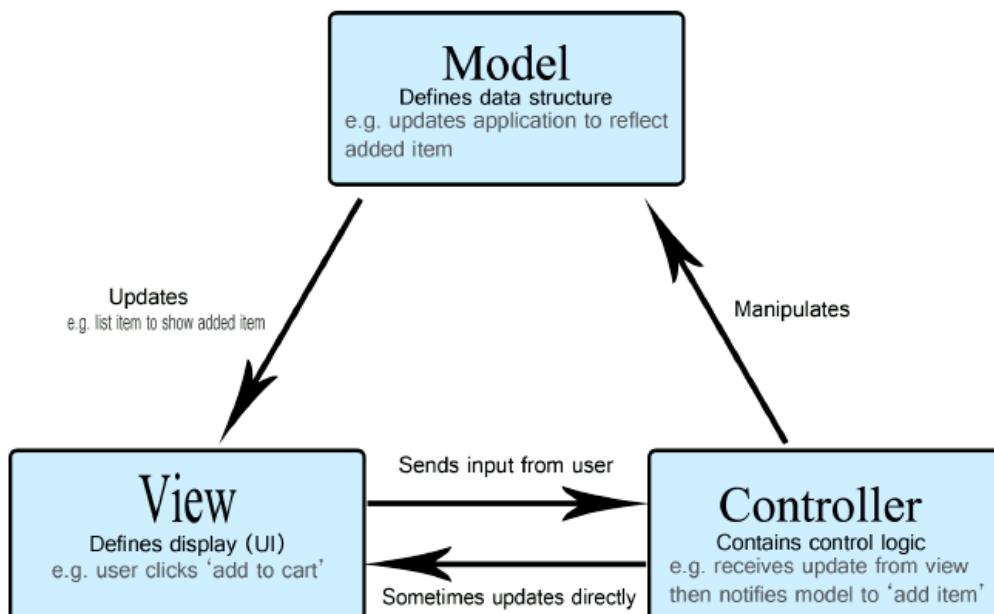


Figure 4.2.2 Model-View-Controller Pattern

4.3 Testing

4.3.1 Test Cases

No.	Test objective	Condition for testsuccess	Expected result	Test result
1.	To test Signup response	Enter valid requirements fields	Successful Signup	As expected
2.	To test Login response	Enter valid email & password	Successful Login	As expected
3.	To test adding products to the Cart response	Choose a product and add to cart	Successful adding product to the cart	As expected
4.	To test checkout of orders response	Select payment method	Successful orders checkout	As expected
5.	To test adding user's address response	Choose an address on the map	Successful adding address	As expected

Table 4.3.1: Test Cases

4.3.2 Types of Testing used

a) Unit Testing:

Unit testing is the process of checking small pieces of code to ensure that the individual parts of a program work properly on their own, speeding up testing strategies and reducing wasted tests.

b) Performance Testing:

Performance testing examines the speed, stability, reliability, scalability, and resource usage of a software application under a specified workload. The focus of

Performance Testing is checking a software program's

- Speed – Determines whether the application responds quickly. For speed, it checks how much time is required to upload a voice or image?
- Scalability – Determines maximum user load the software application can handle. Here, it determines whether the software product is stable in case of varying workloads.

c) Load Testing:

This type of non-functional software testing process determines how the software application behaves while being accessed by multiple users simultaneously. For example, it examines, is it possible for multiple users to access the same application at the same time or not?

d) User Interface Testing:

Type of testing which is performed to check how user-friendly the application is. It is performed by testing teams. Basically, it checks how easily one can explore various features offered by our application?

e) Functional Testing:

Functional testing is a type of software testing that validates the software system against the functional requirements/specifications. The purpose of Functional tests is to test each function of the software application, by providing appropriate input, verifying the output against the Functional requirements. Basically, it checks whether the application is satisfying all the business cases or not?

Conclusion

This App will provide users with modules to help them with their task. The player's major goal in building the app was to prioritize quality above quantity. As previously indicated, more modules and functionality will be introduced soon to make the program more solid and adaptable.

The primary goal and aim of this program are to enable and assist people in using an e-commerce-based platform, which is widely utilized throughout the world for online shopping and purchase of commodities and products nowadays. As a result, individuals may enjoy, conveniently access, and purchase all things through internet shopping and feel fully integrated into society. Our effort will not end here; we will continue to improve the software and provide new features for those who are unable to perceive the world from any APP-Name (completely blind people). We'll aim to include a feature that makes this app totally accessible to blind people; they'll be able to control it with their voice and dress themselves according to their preferences. We'll be adding additional features as time goes on.

References

1. Dart. No date. Dart programming language | Dart. <https://dart.dev/>
2. Flutter. no date. Flutter: Build apps for any screen. <https://flutter.dev/>
3. Google. 2022. Everything you need to build on Android. <https://developer.android.com/studio/features>
 1. AppDividend. no date. Flutter Mobile Development Guide for Beginners.
<https://appdividend.com/category/flutter>
 2. Abilitynet. 2013. How to use TalkBack in Android Jelly Bean.
<https://mcmw.abilitynet.org.uk/talkback-android-jelly-bean-4-2-2>
 3. Android Accessibility Help. 2022. Get started on Android with TalkBack.
<https://support.google.com/accessibility/android/answer/6283677?hl=en>
 4. AppDividend. no date. Flutter Mobile Development Guide For Beginners.
<https://appdividend.com/category/flutter/>

5. Code Carbon. 2020. Major Advantages And Disadvantages Of Dart Language.
<https://codecarbon.com/pros-cons-dart-language/>
6. Collin, C. 2021. 5 Critical Steps for Your Software Implementation Plan.
<https://www.softwareadvice.com/resources/software-implementation-plan/>
7. Darrell, R. 2011. The Future of Shopping. Harvard Business Review. Read on 08.05.2022.
8. <https://hbr.org/2011/12/the-future-of-shopping>
9. Everyday. No date. Essential reads: Where marketing meets strategy - Everyday Online Marketing. <https://everydayonlinemarketing.com/>
10. GeeksforGeeks. No date. A computer science portal for geeks. <https://www.geeksforgeeks.org>
11. James, H. 2022. What is Java? Definition, Meaning & Features of Java Platforms.
<https://www.guru99.com/java-platform.html>
12. Lockwood, B. no date. 10 Technology Challenges. <https://sgrlaw.com/ttl-articles/10-technology-challenges/>
13. Michael Aldrich Archive. No date. Welcome to the Michael Aldrich Archive.
<https://www.aldricharchive.co.uk/>
14. MonsieurZbanowanYY. 2022. E-Commerce-App-UI-Flutter.
<https://github.com/MonsieurZbanowanYY/E-Commerce-App-UI-Flutter>
15. Nazmun Nessa Moon , Shaheena Sultana, Fernaz Narin Nur & Mohd Saifuzzaman. 2017. A Literature Review of the Trend of Electronic Commerce in Bangladesh Perspective.
https://globaljournals.org/GJMBR_Volume17/3-A-Literature-Review-of-the-Trend.pdf
16. Oberlo. 2022. FULFILL YOUR DROPSHIPPING DREAMS. <https://www.oberlo.com/>
17. Pravine, V. no date. A 5-STEP PLAN FOR SOFTWARE IMPLEMENTATION SUCCESS.
<https://www.chetu.com/blogs/technical-perspectives/softwareimplementation-plan.php>
18. Richard F. Schmidt. 2013. Chapter 19 - Software Implementation
<https://www.sciencedirect.com/science/article/pii/B9780124077683000197>
19. Salesforce, Inc. 2022. CRM Software; Cloud Computing Solutions - Salesforce EMEA.
<https://www.salesforce.com/eu/>
20. Santex. No Date. 7 Steps for Software Implementation Success. <https://santexgroup.com/blog/7-steps-for-software-implementation-success/>
21. Shopify. No Date. Shopify: Start and grow your e-commerce business. <https://www.shopify.com>
22. The Flutter way. 2020. Online Shop App - Flutter UI - Speed Code. Published on 03.07.2020. <https://www.youtube.com/watch?v=XBKzpTz65Io>
23. Unsplash . no date. Beautiful Free Images & Pictures | Unsplash. <https://unsplash.com/>

24. Wikipedia contributors. 2022. Model–view–Controller. In Wikipedia, The Free Encyclopedia.
<https://en.wikipedia.org/wiki/Model%20view%20controller>
25. IBM Cloud Education. 2021. REST APIs. WWW document. Available at:
<https://www.ibm.com/cloud/learn/rest-apis>