

CAN THO UNIVERSITY
COLLEGE OF INFORMATION AND COMMUNICATION
TECHNOLOGY



PROJECT - FUNDAMENTAL REPORT
INFORMATION TECHNOLOGY

Topic

**GROCERY
MANAGEMENT APPLICATION
GROCERYPOS**

Author: Nguyen Thanh Phat
Student code: B2005853
Code year: K46

Cantho, 02/2023

CAN THO UNIVERSITY
COLLEGE OF INFORMATION AND COMMUNICATION
TECHNOLOGY



PROJECT - FUNDAMENTAL REPORT
INFORMATION TECHNOLOGY

Topic

GROCERY
MANAGEMENT APPLICATION
GROCERYPOS

Instructor: Nguyen Thanh Phat
Dr. Lam Nhut Khang Author: Nguyen Thanh Phat
 Student code: B2005853
 Year code: K46

Cantho, 02/2023

Acknowledge

Contents

I	Introduction	2
1	Background	3
2	Problem statement	3
3	Aims and objectives	3
4	Research objects and scope	3
4.1	Research Objectives:	3
4.2	Research Scope:	4
5	Solution approach	4
6	Summary of contributions and achievements	4
7	Organization of the report	4
II	Content	6
1	Requirements and specification	7
1.1	Overall Description	7
1.1.1	Product Perspective	7
1.1.2	Product Functions	7
1.1.3	User Classes and Characteristics	9
1.1.4	Operating Environment	10
1.2	Requirements Specification	10
1.2.1	External Interface Requirements	10
1.2.2	Functional Requirement	10
1.2.2.1	Login	10
1.2.2.2	Sign up	11
1.2.2.3	Log out	12
1.2.2.4	Manage Stock	13
1.2.2.5	Manage Products	14
1.2.2.6	Manage Categories	15
1.2.2.7	Create invoice	16
1.2.2.8	Manage Invoices	17
1.2.2.9	Manage Revenue and Expenditure	18
1.2.2.10	Print Invoices	19
1.2.2.11	Manage Suppliers	20
1.2.2.12	Manage Customers	21
1.2.2.13	Manage Store	22

1.2.3	Nonfunctional Requirements	23
2	Software Design	24
2.1	Application Architecture	24
2.1.1	Architectural Design	24
2.1.2	Decomposition Description	25
2.2	Data Design	25
2.2.1	Data Description	25
2.2.1.1	Entity-Relationship Diagram	25
2.2.1.2	Firebase Database Diagram	26
2.2.2	Data Dictionary	30
2.3	Detailed Design	33
2.3.1	User Authentication	33
2.3.1.1	Login View	33
2.3.1.2	Sign Up View	34
2.3.2	Home View	35
2.3.3	User Profile View	37
2.3.4	Store Profile View	38
2.3.5	Contacts	39
2.3.5.1	Supplier View	39
2.3.5.2	Customer View	41
2.3.6	Inventory View	43
2.3.6.1	Category View	43
2.3.6.2	Product View	45
2.3.7	Point of Sales	47
3	Implementation	49
3.1	User Authentication	50
3.1.1	Login View	50
3.1.2	Sign Up View	51
3.2	Home View	52
3.3	User Profile View	54
3.4	Store Profile View	55
3.5	Contacts	56
3.5.1	Supplier View	56
3.5.2	Customer View	58
3.6	Inventory View	60
3.6.1	Category View	60
3.6.2	Product View	62
3.7	Point of Sales	64
4	Testing	67
4.1	Introduction	67
4.1.1	Testing objectives	67
4.1.2	Testing scope	67
4.2	Evaluation of Testing Results	67

III Conclusion

68

List of Figures

1.1	General Usecase	9
1.2	UC01a.Login	10
1.3	UC01b.Sign up	11
1.4	UC01c.Log out	12
1.5	UC02a.Manage Stock	13
1.6	UC02b.Manage Products	14
1.7	UC02c.Manage Categories	15
1.8	UC03a.Create Invoice	16
1.9	UC03b.Manage Invoices	17
1.10	UC03c.Manage Revenue	18
1.11	UC03d.Print Invoice	19
1.12	UC04a.Manage Suppliers	20
1.13	UC04b.Manage Customers	21
1.14	UC04c.Manage Store	22
2.1	Application Architecture	24
2.2	Decomposition Description	25
2.3	Entity-Relationship Diagram	25
2.4	Firestore General	26
2.5	Firestore: User Collections	26
2.6	Firestore: Store Profile Collections	27
2.7	Firestore: Categories Collections	27
2.8	Firestore: Suppliers Collections	28
2.9	Firestore: Customers Collections	28
2.10	Firestore: Products Collections	29
2.11	Firestore: Invoices Collections	29
2.12	Detail Design: Login View	33
2.13	Detail Design: Sign Up View	34
2.14	Detail Design: Home View	35
2.15	Detail Design: Home View Side Menu	36
2.16	Detail Design: User Profile View	37
2.17	Detail Design: Store Profile View	38
2.18	Detail Design: Supplier List View	39
2.19	Detail Design: Supplier Entry View	40
2.20	Detail Design: Customer List View	41
2.21	Detail Design: Customer Entry View	42

2.22 Detail Design: Category List View	43
2.23 Detail Design: Category Entry View	44
2.24 Detail Design: Product List View	45
2.25 Detail Design: Product Entry View	46
2.26 Detail Design: Invoice List View	47
2.27 Detail Design: Invoice Entry View	48
3.1 Implementation: Login View	50
3.2 Implementation: Sign Up View	51
3.3 Implementation: Home View	52
3.4 Implementation: Home View Side Menu	53
3.5 Implementation: User Profile View	54
3.6 Implementation: Store Profile View	55
3.7 Implementation: Supplier List View	56
3.8 Implementation: Supplier Entry View	57
3.9 Implementation: Customer List View	58
3.10 Implementation: Customer Entry View	59
3.11 Implementation: Category List View	60
3.12 Implementation: Category Entry View	61
3.13 Implementation: Product List View	62
3.14 Implementation: Product Entry View	63
3.15 Implementation: Invoice List View	64
3.16 Implementation: Invoice Entry View - First	65
3.17 Implementation: Invoice Entry View - Second	66

List of Tables

1.1	User Authentication: Log in	11
1.2	User Authentication: Sign up	12
1.3	User Authentication: Log out	13
1.4	Inventory Management: Manage Stock	14
1.5	Inventory Management: Manage Products	15
1.6	Inventory Management: Manage categories	16
1.7	Sales Management: Create Invoice	17
1.8	Sales Management: Manage Invoices	18
1.9	Sales Management: Manage Revenue and Expenditure	19
1.10	Sales Management: Print Invoices	20
1.11	Store Information Management: Manage Suppliers	21
1.12	Store Information Management: Manage Customers	22
1.13	Store Information Management: Manage Store Information	23
2.1	Data Description: User	30
2.2	Data Description: Category	30
2.3	Data Description: Suppliers	30
2.4	Data Description: Category	31
2.5	Data Description: Product	31
2.6	Data Description: Invoices	31
2.7	Data Description: invoices	32

Abstract

This research investigates the challenges faced by small businesses, specifically small grocery store owners, in managing sales and inventory. The study aims to develop and evaluate a cross-platform mobile application solution using the Flutter library and Google's Firebase technology. The application is designed with a user-friendly interface to assist small businesses in managing sales and inventory, addressing the challenges they face in this area. The scope of the study is limited to small businesses and does not consider larger organizations. The outcomes of this research will contribute to the knowledge of mobile application solutions for small businesses.

Part I

Introduction

1 Background

The rapid development of information technology has led to its widespread use in various fields, including business operations. As such, the integration of software applications into sales business operations has become a vital need to improve management quality and increase revenue. However, choosing a suitable software application on a suitable platform (web, desktop application, mobile application) for business operations remains a difficult issue, particularly for small and medium-sized businesses.

2 Problem statement

Several sales management applications are available in the market that is suitable for typical business owners, such as KiotViet, a sales management software. Some medium-sized stores have also invested in a suitable management model through a website or computer application, along with supporting devices.

Nevertheless, most applications are commercial, which small store owners cannot afford to invest in, along with the necessary equipment. Therefore, they are forced to rely on manual bookkeeping methods to manage their sales.

3 Aims and objectives

The purpose of this project is to develop a software application that integrates with sales business operations to enhance management quality and increase revenue. This project aims to provide a cost-effective solution for small and medium-sized businesses to manage their sales operations. The developed software application will have a user-friendly interface that allows for easy integration and adoption by businesses.

4 Research objects and scope

4.1 Research Objectives:

The research aims to develop and evaluate a mobile application solution for small businesses to assist in managing sales and inventory. The application will be developed using the Flutter framework and will utilize Firebase as the data platform. The specific objectives of the research are to:

- Investigate challenges faced by small businesses in managing sales and inventory.
- Develop a mobile application solution for small businesses using the Flutter framework and Firebase.
- Evaluate the usability and effectiveness of the developed application through user testing and surveys.

The outcomes of the study will contribute to the knowledge of mobile application solutions for small businesses.

4.2 Research Scope:

- The research investigates the retail business processes of small grocery store owners and businesses. The scope of the study is limited to small businesses and will not consider larger organizations.
- A cross-platform mobile application will be developed using the Flutter library and Google's Firebase technology.

5 Solution approach

- Building a cross-platform mobile application using Flutter.
- Researching Firebase technologies including Firebase Authentication, Firestore and Firebase Storage.
- Business processes related to the retail operations of small grocery store owners.

6 Summary of contributions and achievements

- This research provides a solution for managing the retail sales aspect of small businesses through the development of a cross-platform mobile application with a user-friendly interface.
- The application is designed to assist small businesses in managing sales and inventory, addressing the challenges they face in this area.

7 Organization of the report

This report includes the following sections:

Introduction: This section provides an overview of the thesis, including the issues that need to be addressed, the history of problem-solving, the objectives of the thesis, the contribution of the thesis, and the content that will be covered.

Content: This section includes a detailed description of the problem, analysis, functional specification, data design and implementation, interface for the application, and evaluation of software testing. The content section is divided into three chapters.

Chapter 1: Problem description. This chapter provides a detailed description of the problem, including functions, requirements, and theoretical foundation.

Chapter 2: System design. This chapter provides an overview of the system, including the architectural design, data design, functional design, and diagrams to help build the system.

Chapter 3: Implementation. This chapter provides an implementation of the system design.

Chapter 4: Testing and evaluation. This chapter presents the testing plan and management, testing scenarios for the main functions of the system.

Conclusion: This section presents the results achieved, the remaining limitations, and the system's further development.

Part II

Content

Chapter 1

Requirements and specification

1.1 Overall Description

1.1.1 Product Perspective

Information systems in management are being widely applied in all fields of society in general and in business industries in particular. Mobile devices today such as smart-phones and tablets are not only purely personal entertainment devices, but also provide strong support in handling tasks.

For small retail stores, the process of tracking and recording books and documents can be time-consuming and labor-intensive. A tablet or phone can replace traditional notebooks, saving costs, time and effort for the store owner. At the same time, it avoids the loss of important information and updates operational information quickly and accurately.

In practice, such small retail models will have a manager, usually the store owner. Therefore, to facilitate the most effective management possible and to simplify the interface design, the system should have only one agent: the User.

The user has user authentication functions including login, sign up and logout. After authentication, the user can access management functions, specifically:

Inventory management: Manage inventory, manage products and manage categories.

Sales management: Create invoice, manage invoices, print invoices and manage revenue and expenditure.

Store information management: Manage suppliers, manage customers and store information.

1.1.2 Product Functions

1. **User Authentication:** This feature allows the user to securely access the system by providing login and sign up logout functionality.

2. **Inventory Management:** This feature enables the user to effectively manage the store's inventory by providing the following sub-features:
 - **Inventory Tracking:** Allows the user to track the quantity and status of items in stock.
 - **Product Management:** Enables the user to add, edit and delete products in the system.
 - **Category Management:** Allows the user to organize products into categories for easier management.
3. **Sales Management:** This feature facilitates the sales process by providing the following sub-features:
 - **Invoice Creation:** Allows the user to Create invoice for sales transactions.
 - **Invoice Management:** Enables the user to view, edit and delete invoices.
 - **Invoice Printing:** Allows the user to print invoices for record-keeping and customer reference.
 - **Revenue and Expenditure Management:** Enables the user to track and manage the store's revenue and expenditure.
4. **Store Information Management:** This feature enables the user to manage information related to the store by providing the following sub-features:
 - **Supplier Management:** Allows the user to add, edit and delete supplier information.
 - **Customer Management:** Enables the user to add, edit and delete customer information.

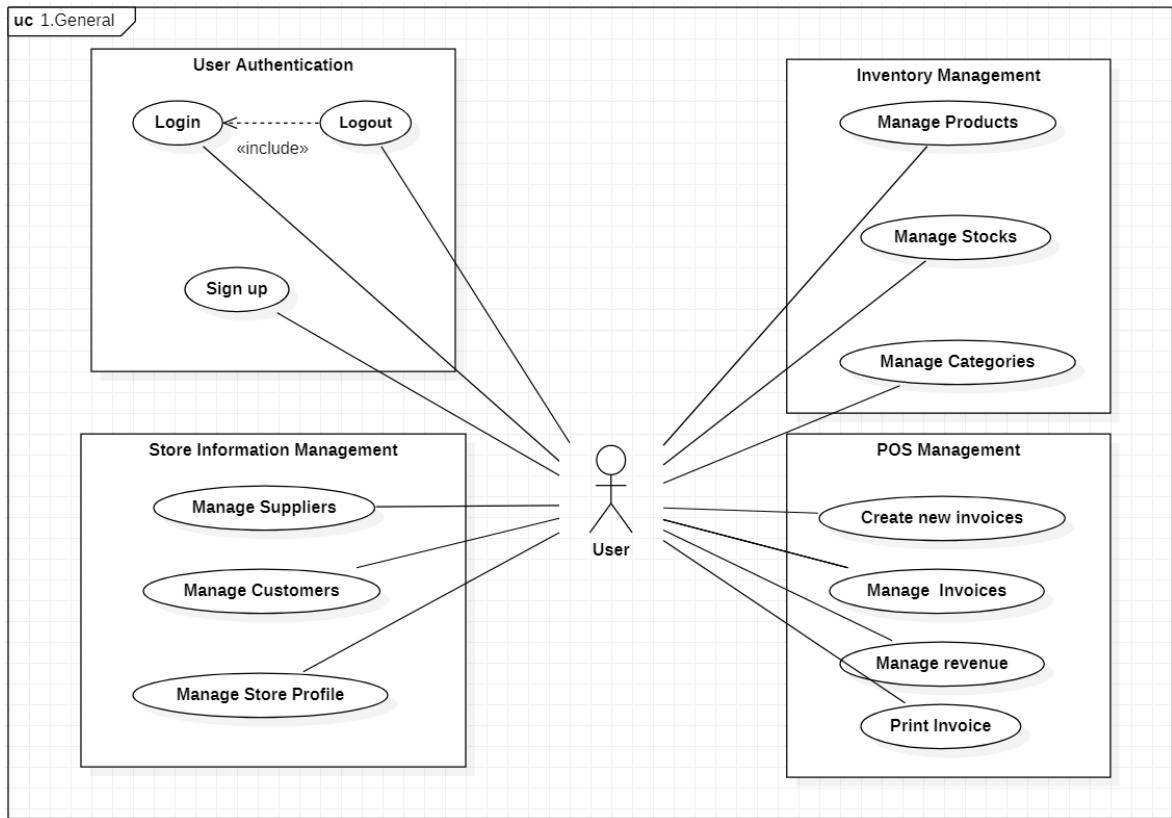


Figure 1.1: General Usecase

1.1.3 User Classes and Characteristics

- **User Class:** Store Owner/Manager
- **Authentication:** The user has the ability to authenticate themselves through login, sign up and logout functions on the mobile application. This ensures that only authorized users have access to the management functions.
- **Inventory Management:** Once authenticated, the user can access the inventory management function on the mobile application which allows them to manage inventory, products, and categories.
- **Sales Management:** The user can also access the sales management function on the mobile application which allows them to create and manage invoices, print invoices, and manage revenue and expenditure.
- **Store Information Management:** The user can access the store information management function on the mobile application which allows them to manage suppliers, customers, and store information.

All information is stored and retrieved from a remote database, allowing for real-time updates and accurate tracking of inventory levels, sales transactions, and store information.

1.1.4 Operating Environment

1.2 Requirements Specification

1.2.1 External Interface Requirements

- **User Interfaces:** The system should have a user-friendly interface on the mobile application for the store owner/manager to access and use the various management functions. The interface should be designed to be easy to navigate and intuitive to use.
- **Hardware Interfaces:** The system should be compatible with common mobile devices such as smartphones and tablets. It should also be able to connect to a remote database server through a network connection.
- **Software Interfaces:** The system should be able to interface with the remote database server to store and retrieve information. It should also be able to interface with other software systems such as accounting or inventory management software if necessary.
- **Communication Interfaces:** The system should be able to communicate with the remote database server through a secure network connection. It should also be able to communicate with other devices or systems if necessary.

1.2.2 Functional Requirement

1.2.2.1 Login

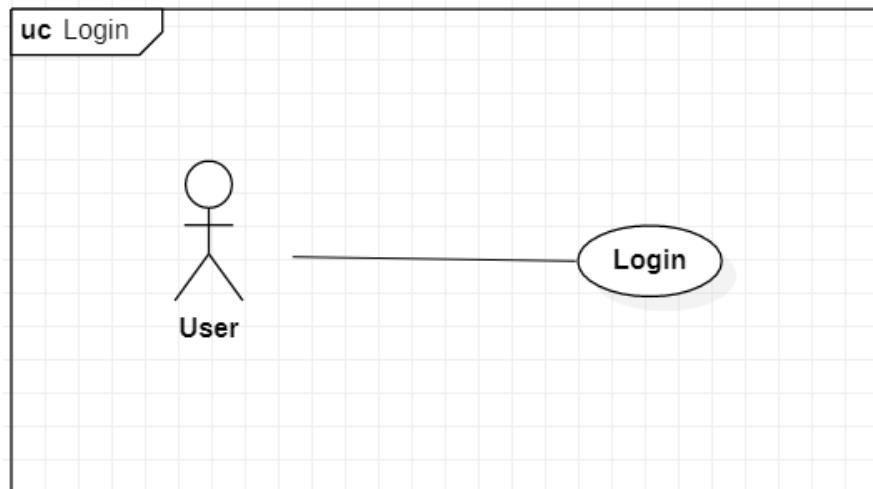


Figure 1.2: UC01a.Login

Use case	User Authentication: Log in
ID	UC01a
Main actor	User
Priority	High
Brief description	Allows the user to securely access the system by providing login functionality.
Trigger	The user opens the application and selects the login option.
Type	Primary
Relationship	
Normal flows	<ol style="list-style-type: none"> 1. The user opens the application. 2. User enters email and password 3. The system verifies the credentials. 4. The user is granted access to the system. 5. The user can access additional functions such as inventory management, sales management and store information management. 6. End login event
Subflows	
Exceptional flow	<ol style="list-style-type: none"> 1. The user enters incorrect login credentials. 2. The system displays an error message and prompts the user to try again.

Table 1.1: User Authentication: Log in

1.2.2.2 Sign up

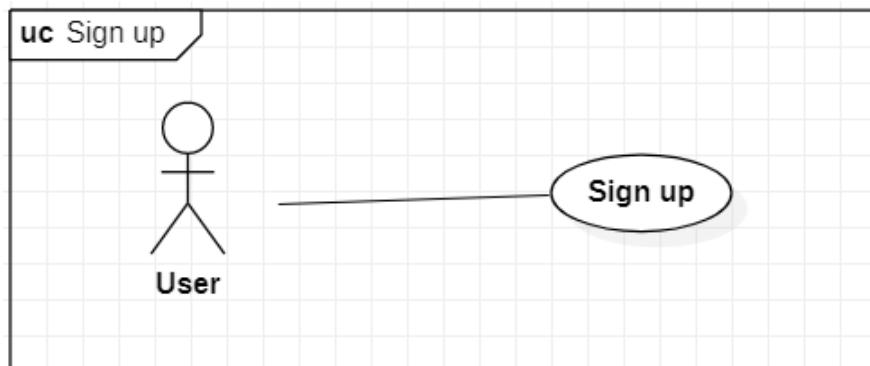


Figure 1.3: UC01b.Sign up

Use case	User Authentication: Sign up
ID	UC01b
Main actor	User
Priority	High
Brief description	Allows the user to create an account in the system.
Trigger	The user opens the application to create an account.
Type	Primary
Relationship	
Normal flows	<ol style="list-style-type: none"> 1. The user opens the application. 2. User enters required information: Name, Address and User credentials User's Email/Phone and Password. 3. The system verifies the credentials. 4. System verifies information and creates a new account for the user. 5. The user is granted access to the system. 6. The user can access additional functions such as inventory management, sales management and store information management. 7. End sign up event
Subflows	
Exceptional flow	<ol style="list-style-type: none"> 1. System error while accessing sign up feature. 2. The system displays an error message and prompts the user to try again.

Table 1.2: User Authentication: Sign up

1.2.2.3 Log out

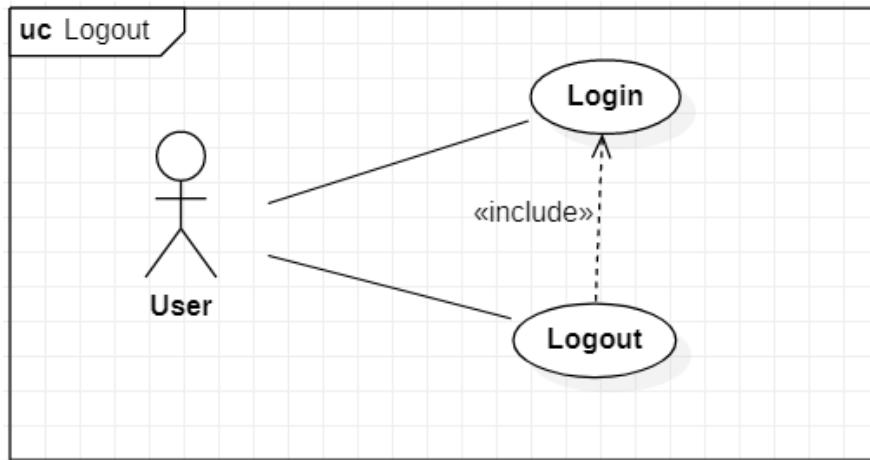


Figure 1.4: UC01c.Log out

Use case	User Authentication: Logout
ID	UC01c
Main actor	User
Priority	High
Brief description	Allows the user to securely log out of the system.
Trigger	User wants to log out of the system.
Type	Primary
Relationship	
Normal flows	<ol style="list-style-type: none"> 1. User logs into the system (UC01a) 2. User navigates to logout feature. 3. System logs user out of the system. 4. User is no longer able to access the system until they log in again. 5. End log out event
Subflows	
Exceptional flow	<ol style="list-style-type: none"> 1. The system displays an error message and prompts the user to try again.

Table 1.3: User Authentication: Log out

1.2.2.4 Manage Stock

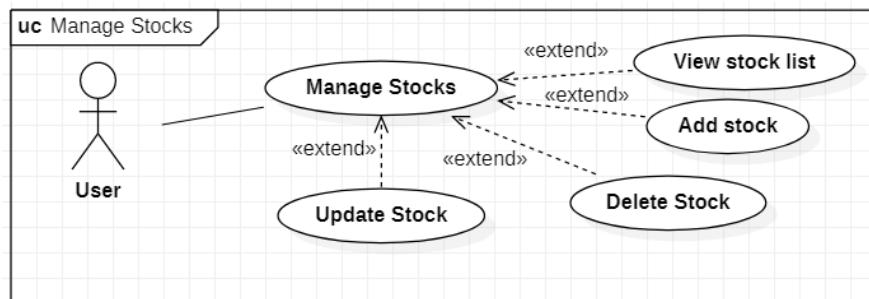


Figure 1.5: UC02a.Manage Stock

Use case	Inventory Management: Manage Stock
ID	UC02a
Main actor	User
Priority	High
Brief description	Allows the user to track the quantity and status of items in stock.
Trigger	User wants to track inventory.
Type	Primary
Relationship	Include UC01a
Normal flows	<ol style="list-style-type: none"> 1. User logs into the system (UC01a). 2. User navigates to inventory tracking feature. 3. System displays inventory tracking interface. 4. User views and updates inventory information as needed. 5. End of use case.
Subflows	
Exceptional flow	<ol style="list-style-type: none"> 1. System error while accessing inventory tracking feature.

Table 1.4: Inventory Management: Manage Stock

1.2.2.5 Manage Products

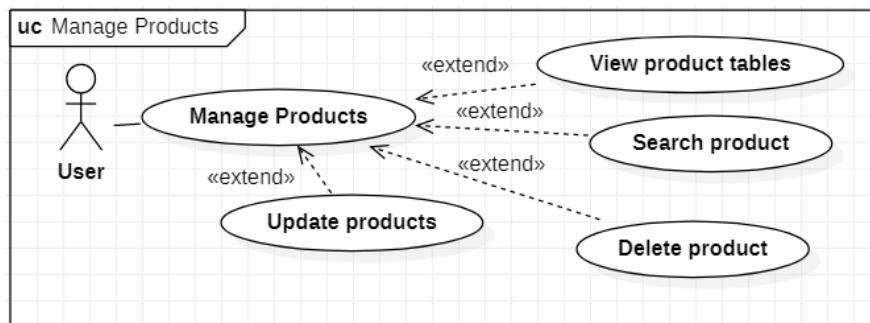


Figure 1.6: UC02b. Manage Products

Use case	Inventory Management: Manage Products
ID	UC02b
Main actor	User
Priority	High
Brief description	Allows the user to add, edit and delete products in the system.
Trigger	User wants to track inventory.
Type	Primary
Relationship	Include UC01a
Normal flows	<ol style="list-style-type: none"> 1. User logs into the system (UC01a). 2. User navigates to product management feature. 3. System displays product management interface. 4. User adds, edits or deletes products as needed. 5. End of use case.
Subflows	
Exceptional flow	<ol style="list-style-type: none"> 1. System error while accessing product management feature.

Table 1.5: Inventory Management: Manage Products

1.2.2.6 Manage Categories

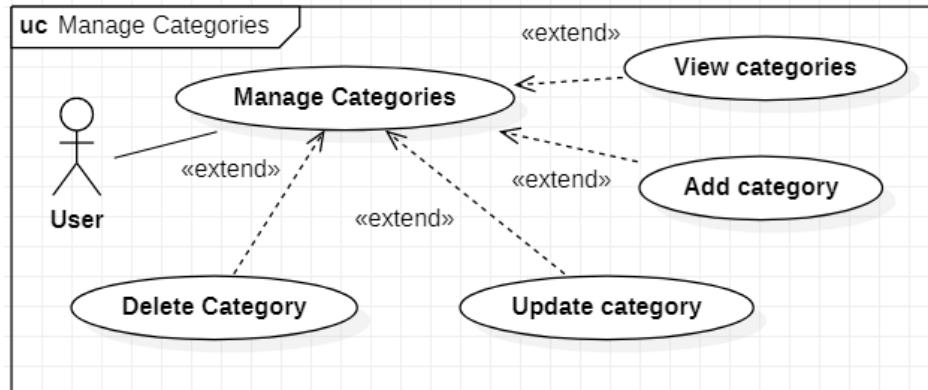


Figure 1.7: UC02c.Manage Categories

Use case	Inventory Management: Manage Categories
ID	UC02c
Main actor	User
Priority	High
Brief description	Allows the user to add, edit and delete categories in the system.
Trigger	User wants to track inventory.
Type	Primary
Relationship	Include UC01a
Normal flows	<ol style="list-style-type: none"> 1. User logs into the system (UC01a). 2. User navigates to product category management. 3. System displays category management interface. 4. User adds, edits or deletes categories as needed. 5. End of use case.
Subflows	
Exceptional flow	<ol style="list-style-type: none"> 1. System error while accessing category management feature.

Table 1.6: Inventory Management: Manage categories

1.2.2.7 Create invoice

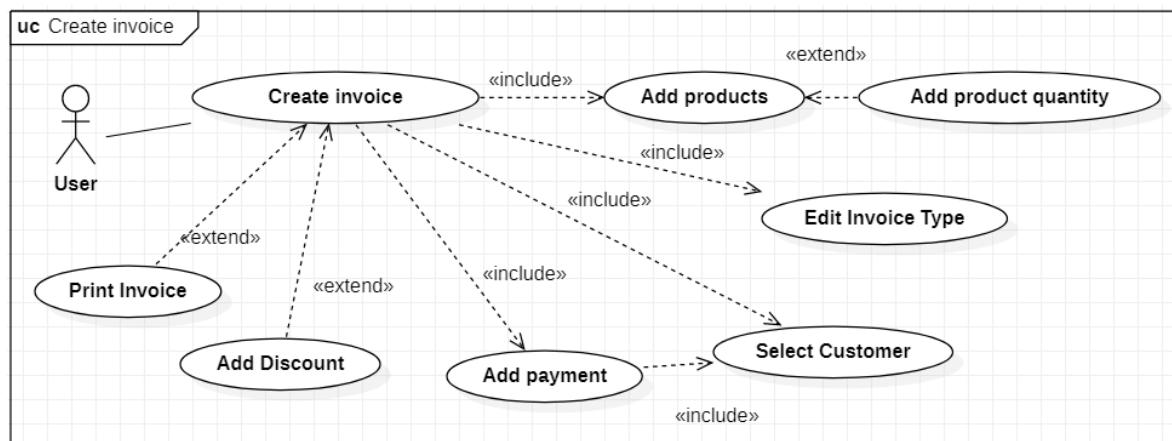


Figure 1.8: UC03a.Create Invoice

Use case	Sales Management: Create invoice
ID	UC03a
Main actor	User
Priority	High
Brief description	Allows the user to Create invoice for sales transactions.
Trigger	User wants to create an invoice.
Type	Primary
Relationship	Include UC01a
Normal flows	<ol style="list-style-type: none"> 1. User logs into the system (UC01a). 2. User navigates to invoice creation feature. 3. System displays invoice creation interface. 4. User input required products and product quantity. Then input customer if invoice is order and option discount. 5. End of use case.
Subflows	<ol style="list-style-type: none"> 1. User adds products to the invoice and update the quantity of products as needed. 2. Select invoice type. 3. User enters customer information. 4. User applies discounts or promotions. 5. Add full payment if retail or add customer payment if customer invoices
Exceptional flow	<ol style="list-style-type: none"> 1. System error while accessing invoice creation feature.

Table 1.7: Sales Management: Create Invoice

1.2.2.8 Manage Invoices

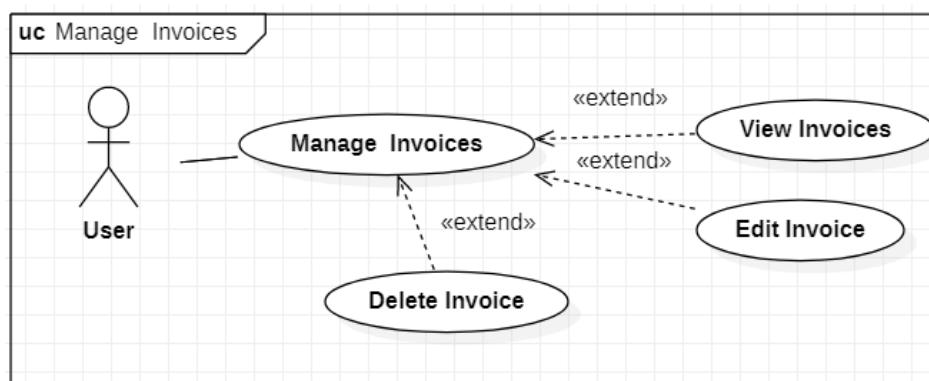


Figure 1.9: UC03b.Manage Invoices

Use case	Sales Management: Manage invoices
ID	UC03b
Main actor	User
Priority	High
Brief description	Enables the user to view, edit and delete invoices. User wants to manage invoices.
Trigger	User wants to track inventory.
Type	Primary
Relationship	Include UC01a, UC03a
Normal flows	<ol style="list-style-type: none"> 1. User logs into the system (UC01a). 2. User navigates to invoice management feature. 3. System displays invoice management interface. 4. User views, edits or deletes invoices as needed. 5. End of use case.
Subflows	
Exceptional flow	<ol style="list-style-type: none"> 1. System error while accessing invoice management feature.

Table 1.8: Sales Management: Manage Invoices

1.2.2.9 Manage Revenue and Expenditure

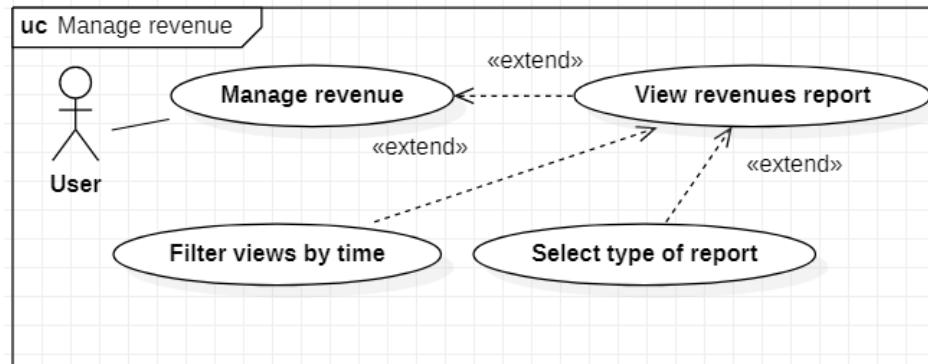


Figure 1.10: UC03c.Manage Revenue

Use case	Sales Management: Manage Revenue and Expenditure
ID	UC03c
Main actor	User
Priority	High
Brief description	Enables the user to track and manage the store's revenue and expenditure.
Trigger	User wants to manage revenue and expenditure.
Type	Primary
Relationship	Include UC01a
Normal flows	<ol style="list-style-type: none"> 1. User logs into the system (UC01a). 2. User navigates to revenue and expenditure management feature. 3. System displays revenue and expenditure management interface. 4. User views and updates views revenue and expenditure information as needed. 5. End of use case.
Subflows	
Exceptional flow	<ol style="list-style-type: none"> 1. System error while accessing revenue and expenditure management feature.

Table 1.9: Sales Management: Manage Revenue and Expenditure

1.2.2.10 Print Invoices

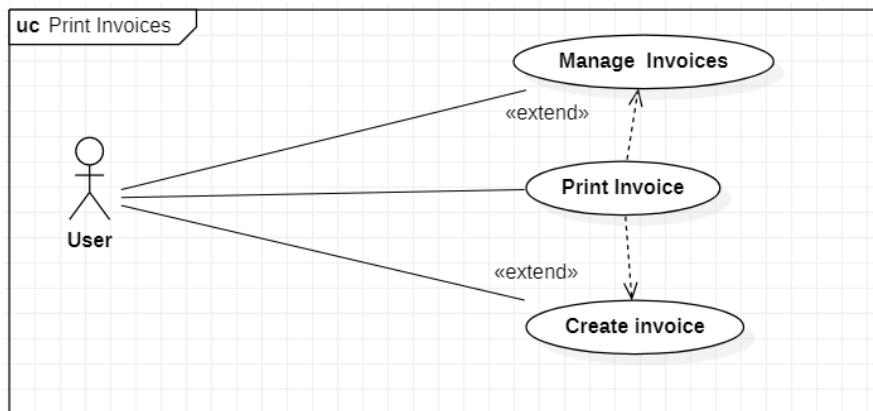


Figure 1.11: UC03d.Print Invoice

Use case	Sales Management: Print Invoices
ID	UC03d
Main actor	User
Priority	High
Brief description	Allows the user to print invoices for record-keeping and customer reference.
Trigger	User wants to print an invoice.
Type	Primary
Relationship	Include UC01a, Extend UC03a , UC03b
Normal flows	<ol style="list-style-type: none"> 1. User logs into the system (UC01a). 2. User navigates to invoice printing feature from UC03a , UC03b. 3. System displays invoice printing interface. 4. User selects printer and modify printing formats as needed. 5. System prints the selected invoice. 6. End of use case.
Subflows	
Exceptional flow	<ol style="list-style-type: none"> 1. System error while accessing printer or printing feature.

Table 1.10: Sales Management: Print Invoices

1.2.2.11 Manage Suppliers

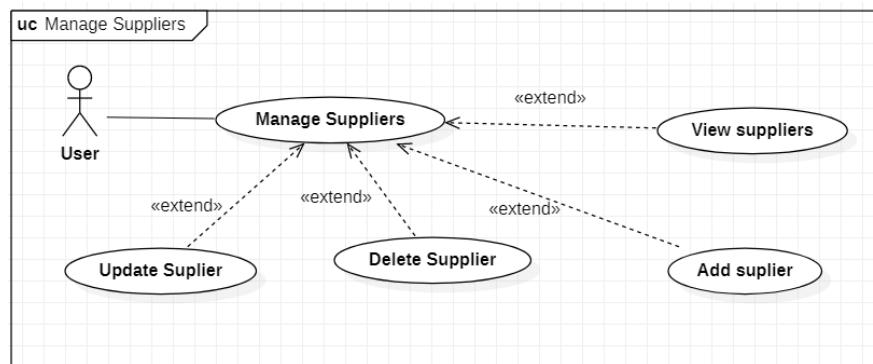


Figure 1.12: UC04a.Manage Suppliers

Use case	Store Information Management: Manage Suppliers
ID	UC04a
Main actor	User
Priority	High
Brief description	Allows the user to add, edit and delete supplier information.
Trigger	User wants to manage supplier information.
Type	Primary
Relationship	Include UC01a
Normal flows	<ol style="list-style-type: none"> 1. User logs into the system (UC01a). 2. User navigates to supplier management feature. 3. System displays supplier management interface. 4. User adds, edits or deletes supplier information as needed. 5. End of use case.
Subflows	
Exceptional flow	<ol style="list-style-type: none"> 1. System error while accessing supplier management feature

Table 1.11: Store Information Management: Manage Suppliers

1.2.2.12 Manage Customers

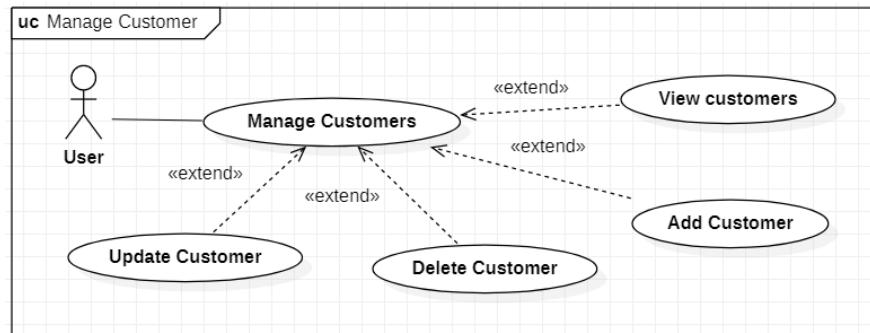


Figure 1.13: UC04b.Manage Customers

Use case	Store Information Management: Manage Customers
ID	UC04b
Main actor	User
Priority	High
Brief description	Allows the user to add, edit and delete customer information.
Trigger	User wants to manage customer information.
Type	Primary
Relationship	Include UC01a
Normal flows	<ol style="list-style-type: none"> 1. User logs into the system (UC01a). 2. User navigates to customer management feature. 3. System displays customer management interface. 4. User adds, edits or deletes customer information as needed. 5. End of use case.
Subflows	
Exceptional flow	<ol style="list-style-type: none"> 1. System error while accessing customer management feature

Table 1.12: Store Information Management: Manage Customers

1.2.2.13 Manage Store

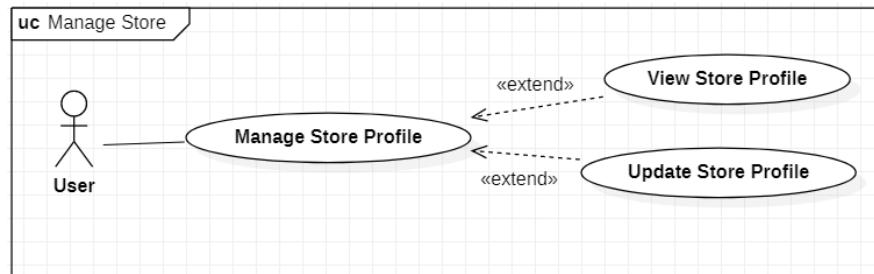


Figure 1.14: UC04c.Manage Store

Use case	Sales Management: Print Invoices
ID	UC04c
Main actor	User
Priority	High
Brief description	Allows the user to edit basic store information such as store name, address and contact.
Trigger	User wants to edit basic store information.
Type	Primary
Relationship	Include UC01a
Normal flows	<ol style="list-style-type: none">1. User logs into the system (UC01a).2. User navigates to store information feature.3. System displays store information interface.4. User edits basic store information such as store name, address and contact.5. End of use case.
Subflows	
Exceptional flow	<ol style="list-style-type: none">1. System error while accessing store information feature.

Table 1.13: Store Information Management: Manage Store Information

1.2.3 Nonfunctional Requirements

Chapter 2

Software Design

2.1 Application Architecture

2.1.1 Architectural Design

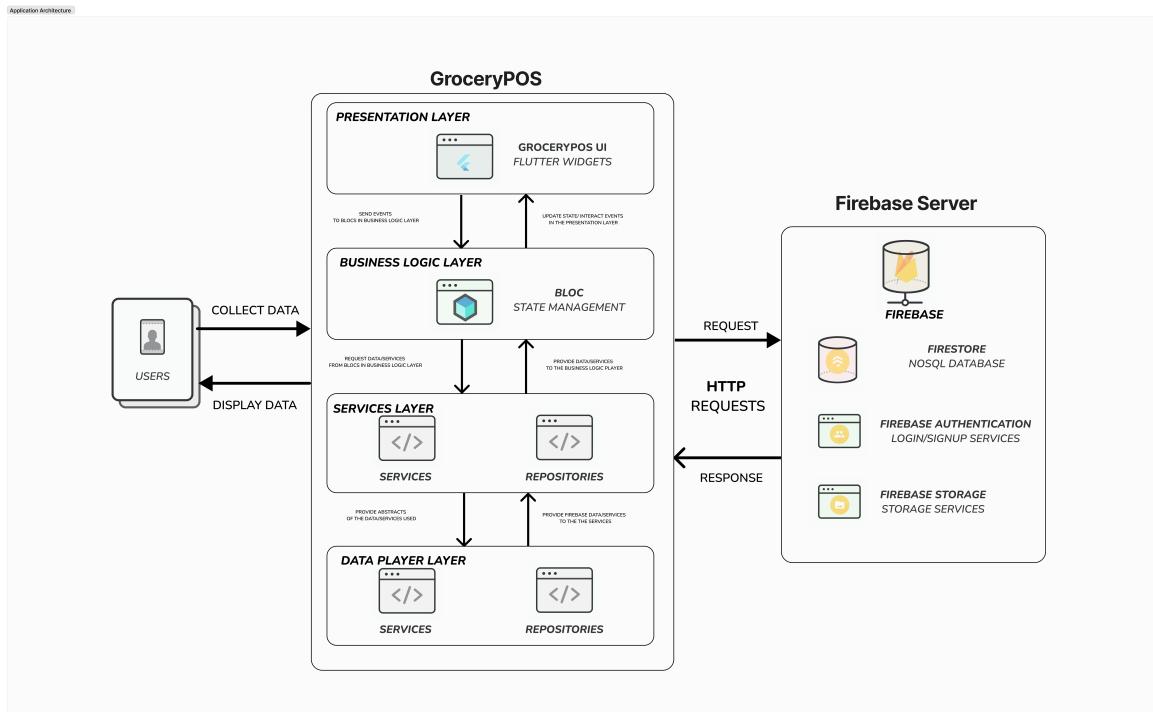


Figure 2.1: Application Architecture

2.1.2 Decomposition Description

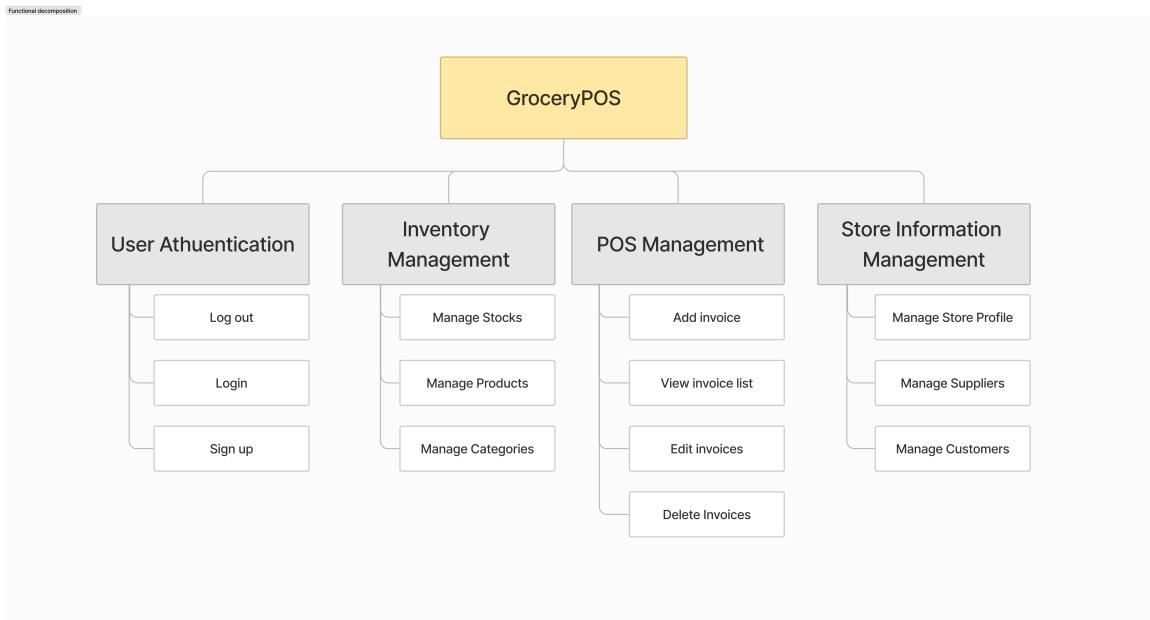


Figure 2.2: Decomposition Description

2.2 Data Design

2.2.1 Data Description

2.2.1.1 Entity-Relationship Diagram

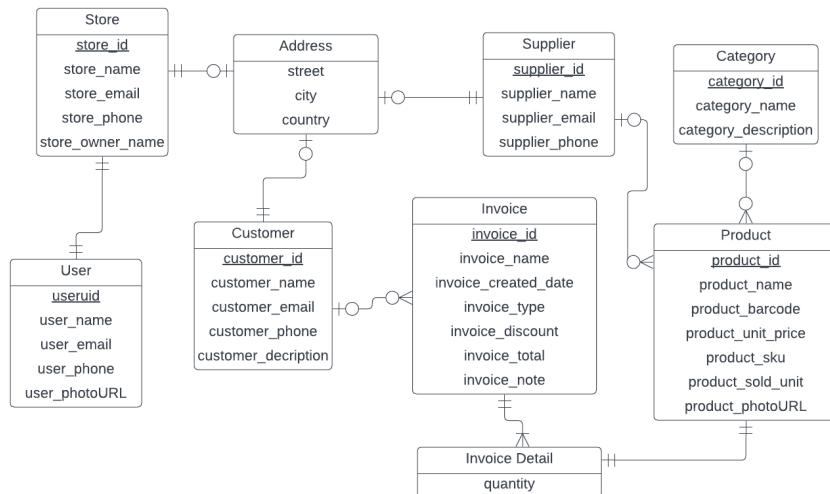


Figure 2.3: Entity-Relationship Diagram

2.2.1.2 Firebase Database Diagram

The root of Firebase Firestore Database:

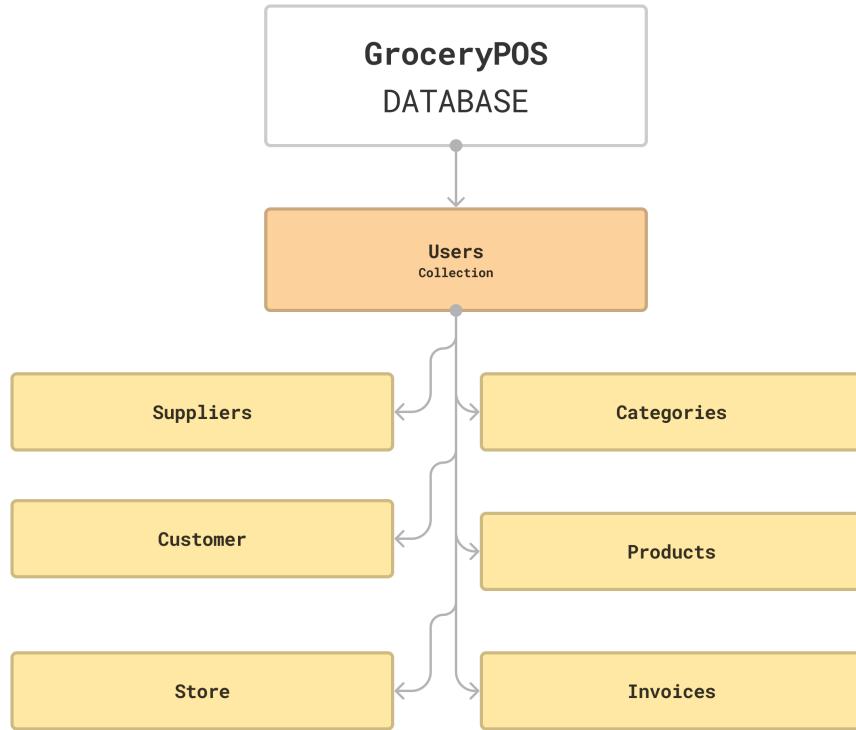


Figure 2.4: Firestore General

The User collections:

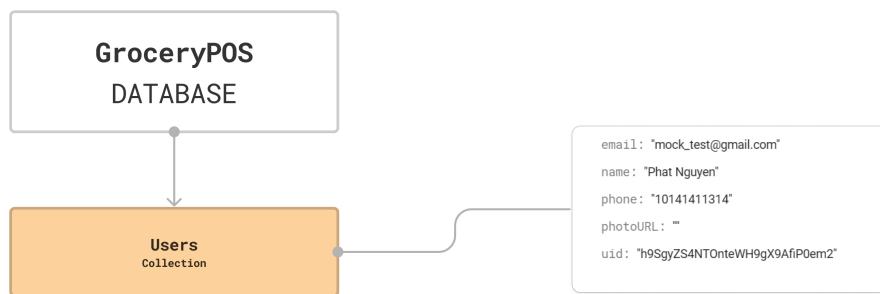


Figure 2.5: Firestore: User Collections

The Subcollections:

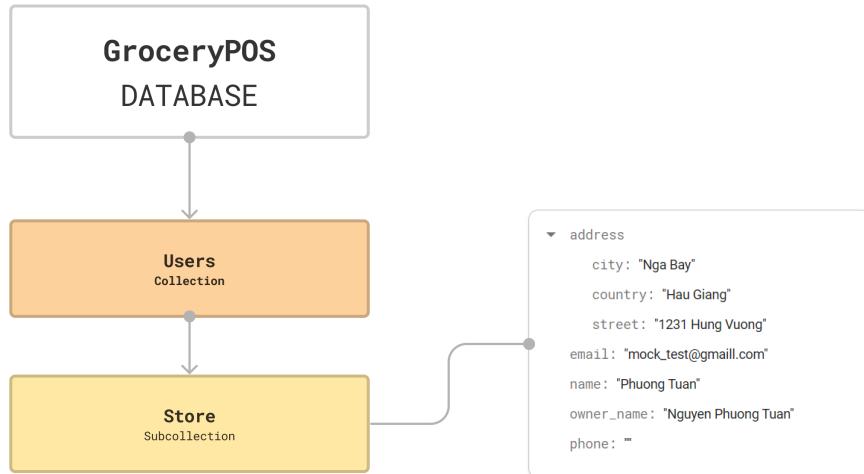


Figure 2.6: Firestore: Store Profile Collections

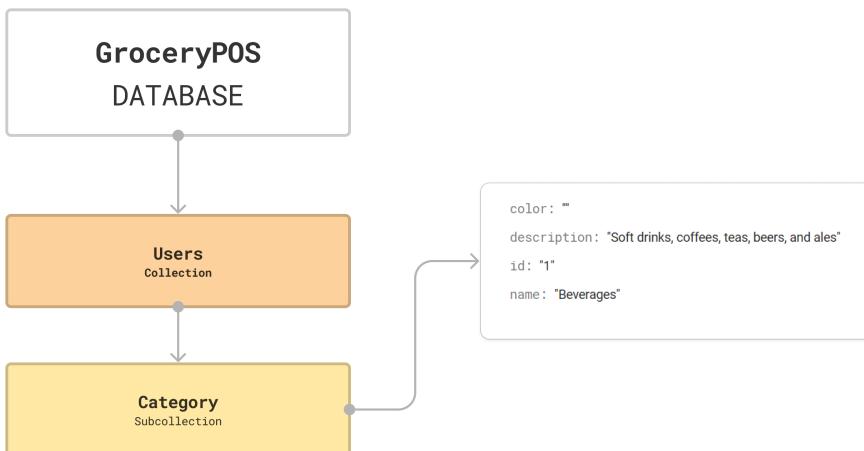


Figure 2.7: Firestore: Categories Collections

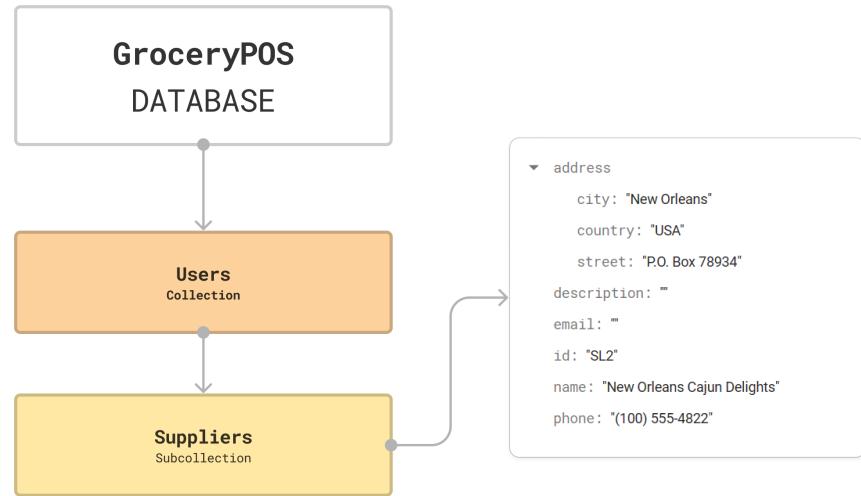


Figure 2.8: Firestore: Suppliers Collections

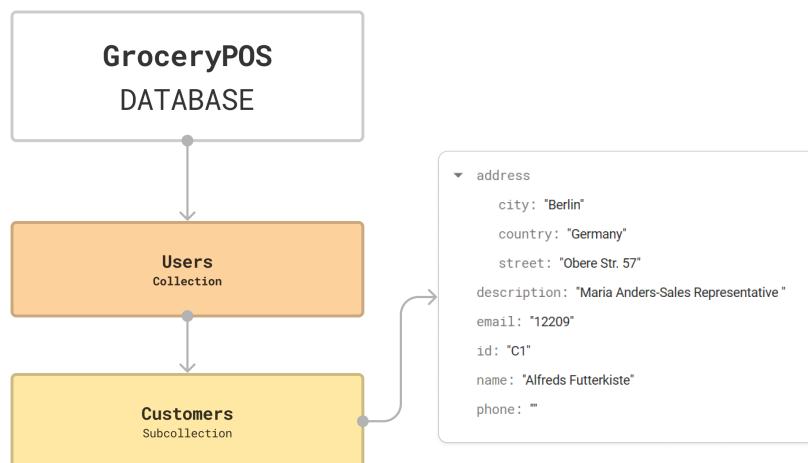


Figure 2.9: Firestore: Customers Collections

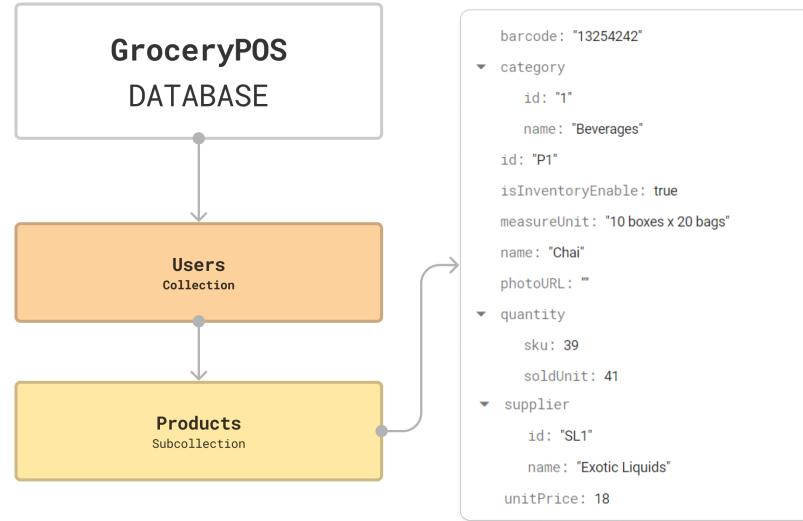


Figure 2.10: Firestore: Products Collections

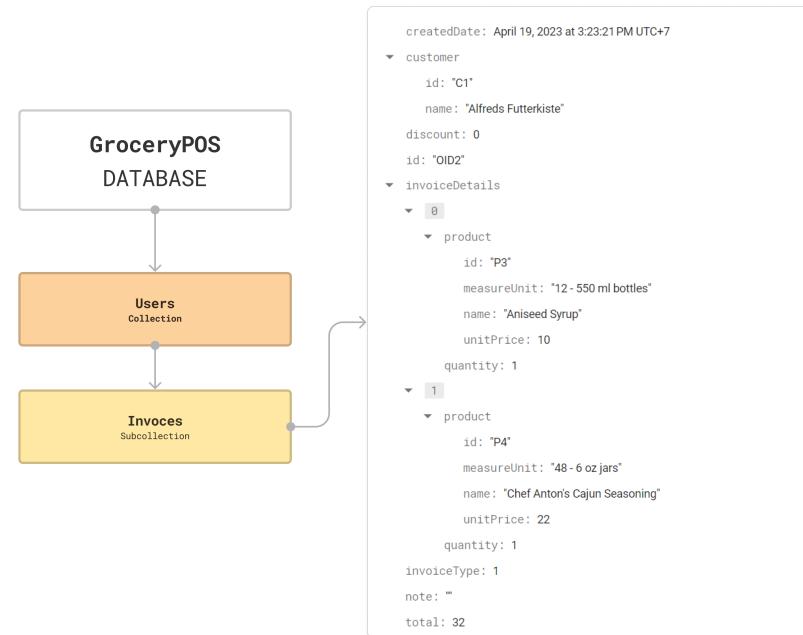


Figure 2.11: Firestore: Invoices Collections

2.2.2 Data Dictionary

Data	Type	Description
user_uid	String	The user authentication auto-generated ID by Firebase Authentication
name	String	The fullname of the user in profile.
email	String	The email which provided for the User Authentication
phone	String	The phone for OTP authentication in register.
photoURL	String	The reference to the user profile's in Google Cloud

Table 2.1: Data Description: User

Data	Type	Description
category_id	String	The unique ID to identify the category
category_name	String	The name of category.
category_description	String	The detail information about category.
category_color	String	The hex code of color to display the category.

Table 2.2: Data Description: Category

Data	Type	Description
supplier_id	String	The unique ID to identify the supplier.
supplier_name	String	The name of suppliers.
supplier_email	String	The email to contact suppliers.
supplier_phone	String	The number to contact suppliers.
supplier_description	String	The detail information about supplier.
supplier_address	Map	The address to retrieve directed contact suppliers
supplier_street	String	The detail of address.
supplier_city	String	The detail of address.
supplier_country	String	The detail of address.

Table 2.3: Data Description: Suppliers

Data	Type	Description
customer_id	String	The unique ID to identify the customer.
customer_name	String	The name of customers.
customer_email	String	The email to contact customers.
customer_phone	String	The number to contact customers.
customer_description	String	The detail information about customer.
customer_address	Map	The address to retrivie directed contact customers
customer_street	String	The detail of address.
customer_city	String	The detail of address.
customer_country	String	The detail of address.

Table 2.4: Data Description: Category

Data	Type	Description
product_id	String	The unique ID to identify the invoices.
product_barcode	String	The barcode of product.
product_name	String	The name of product.
product_measureUnit	String	The means of measurement of product.
product_unitPrice	Number	The price of product per measure unit.
product_photoURL	String	The reference to Google Cloud to retrive photo.
product_quanity	map	The inventory quantity of the product.
product_sku	Number	The number of stock on unit of product
product_soldUnit	Number	The number of sold unit of product
product_supplier	map	The map to use reference Supplier Collections
product_supplier_id	String	The id of Supplier.
product_supplier_name	String	The name of Supplier.
product_category	map	The map to use reference Category Collections
product_category_id	String	The id of Category.
product_category_name	String	The name of Category.

Table 2.5: Data Description: Product

Data	Type	Description
invoice_id	String	The unique ID to identify the invoices.
invoice_createdDate	Timestamp	The timestamp when the invoice created
invoice_discount	Number	The discount of invoice.
invoice_note	String	The note of invoice.
invoice_total		The total cost of invoice.
invoice_invoice_details	Array	The list of invoice details of invoice.
invoice_invoiceType	Number	The number to identify the type of invoice: 0: Retail Invoices, 1: Customer Invoices
invoice_customer	Map	The map to reference Customer Collections
invoice_customer_id	String	The unique ID to identify the customer.
invoice_customer_name	String	The name of customers.

Table 2.6: Data Description: Invoices

Data	Type	Description
product	map	The map object to store a copy of Product details
product_id		The map to use reference Product Collections
product_barcode	String	The barcode of product.
product_name	String	The name of product.
product_measureUnit	String	The means of measurement of product.
proudct_unitPrice	Number	The price of product per measure unit.
quantity	Number	The quantity of product in that invoice

Table 2.7: Data Description: invoices

2.3 Detailed Design

2.3.1 User Authentication

2.3.1.1 Login View

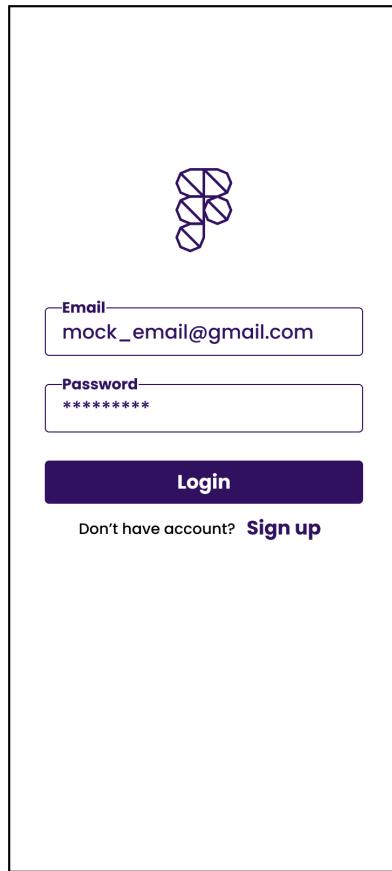


Figure 2.12: Detail Design: Login View

2.3.1.2 Sign Up View

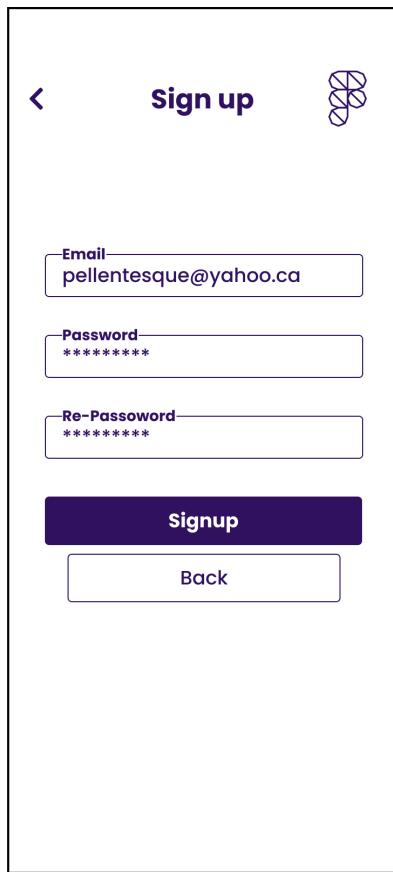


Figure 2.13: Detail Design: Sign Up View

2.3.2 Home View

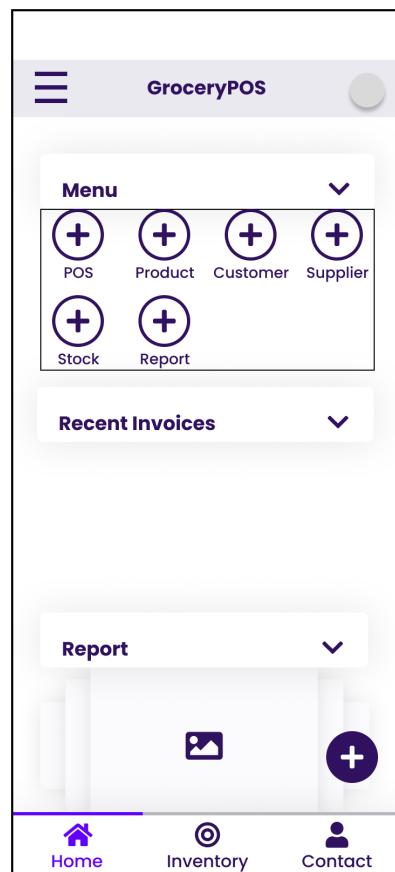


Figure 2.14: Detail Design: Home View

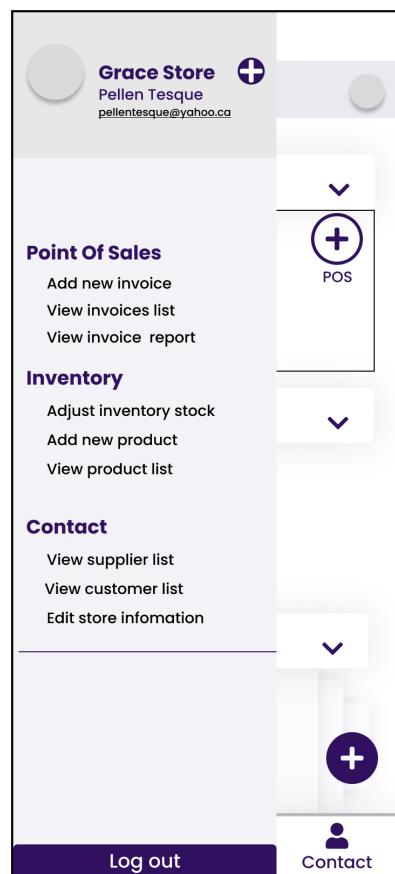


Figure 2.15: Detail Design: Home View Side Menu

2.3.3 User Profile View

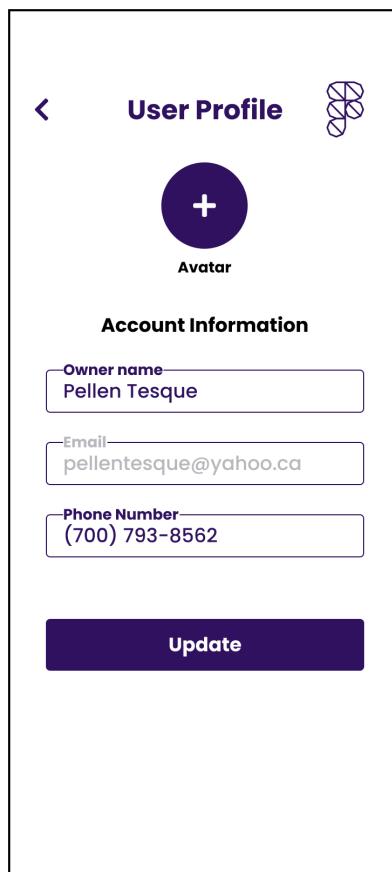


Figure 2.16: Detail Design: User Profile View

2.3.4 Store Profile View

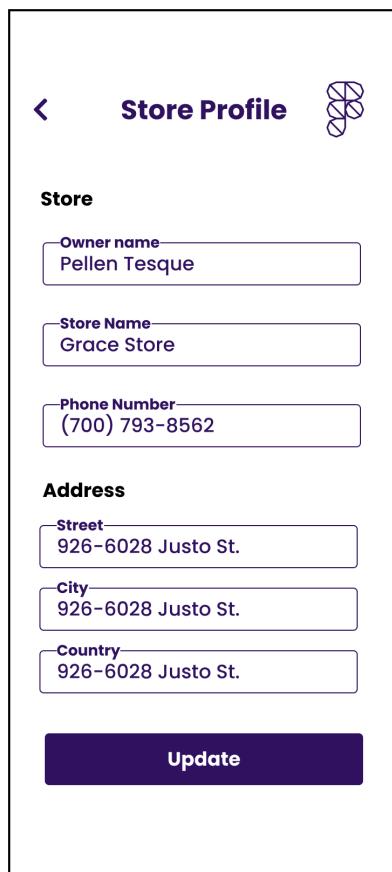


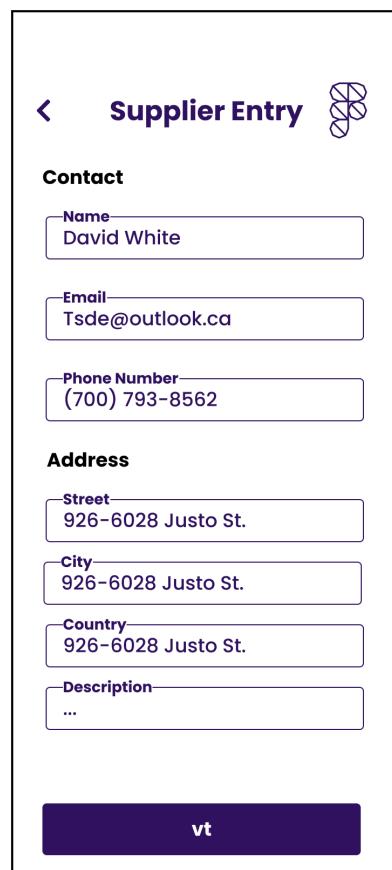
Figure 2.17: Detail Design: Store Profile View

2.3.5 Contacts

2.3.5.1 Supplier View



Figure 2.18: Detail Design: Supplier List View



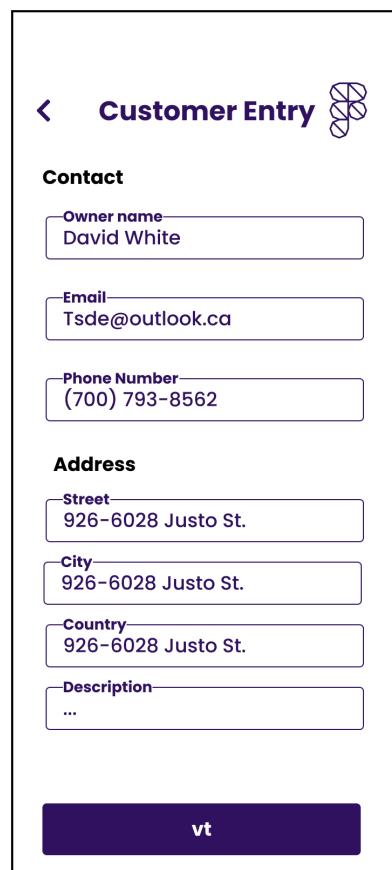
The image shows a mobile-style "Supplier Entry" form. At the top left is a back arrow icon, followed by the title "Supplier Entry" in bold, and a small circular logo with a stylized "S". Below the title is a section labeled "Contact" containing three input fields: "Name" (David White), "Email" (Tsde@outlook.ca), and "Phone Number" ((700) 793-8562). Underneath is a section labeled "Address" with four input fields: "Street" (926-6028 Justo St.), "City" (926-6028 Justo St.), "Country" (926-6028 Justo St.), and "Description" (...). At the bottom right is a large blue button with the white text "vt".

Figure 2.19: Detail Design: Supplier Entry View

2.3.5.2 Customer View



Figure 2.20: Detail Design: Customer List View



The image shows a mobile-style interface titled "Customer Entry" with a logo of two overlapping circles. The screen is divided into sections for "Contact" and "Address".

Contact

- Owner name: David White
- Email: Tsde@outlook.ca
- Phone Number: (700) 793-8562

Address

- Street: 926-6028 Justo St.
- City: 926-6028 Justo St.
- Country: 926-6028 Justo St.
- Description: ...

At the bottom is a large blue button labeled "vt".

Figure 2.21: Detail Design: Customer Entry View

2.3.6 Inventory View

2.3.6.1 Category View

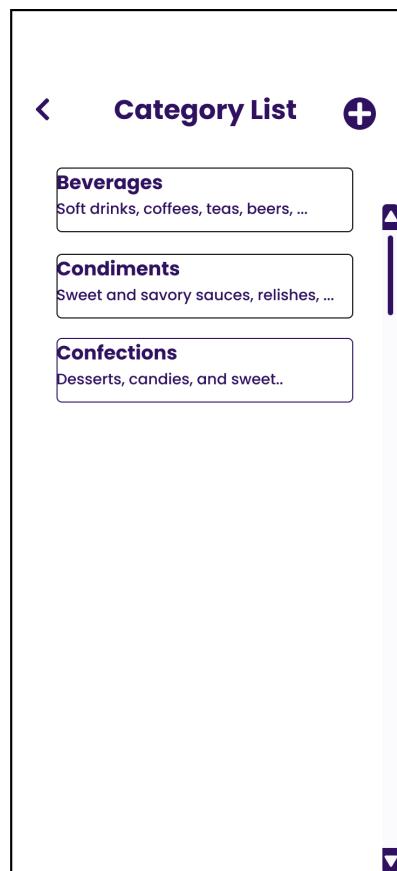


Figure 2.22: Detail Design: Category List View

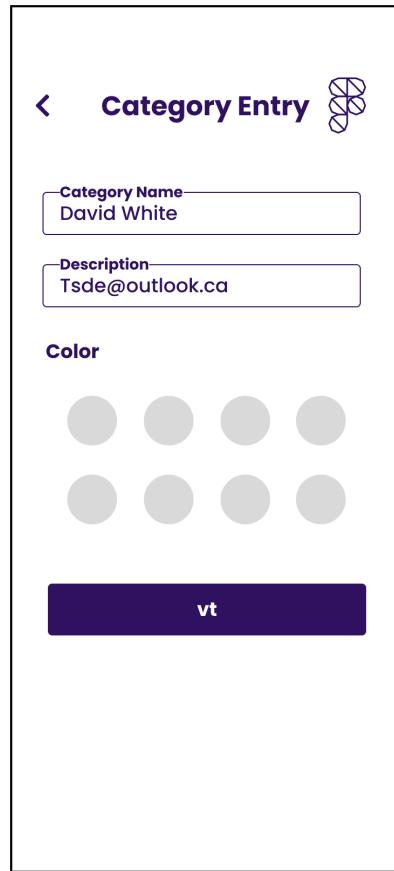


Figure 2.23: Detail Design: Category Entry View

2.3.6.2 Product View



Figure 2.24: Detail Design: Product List View

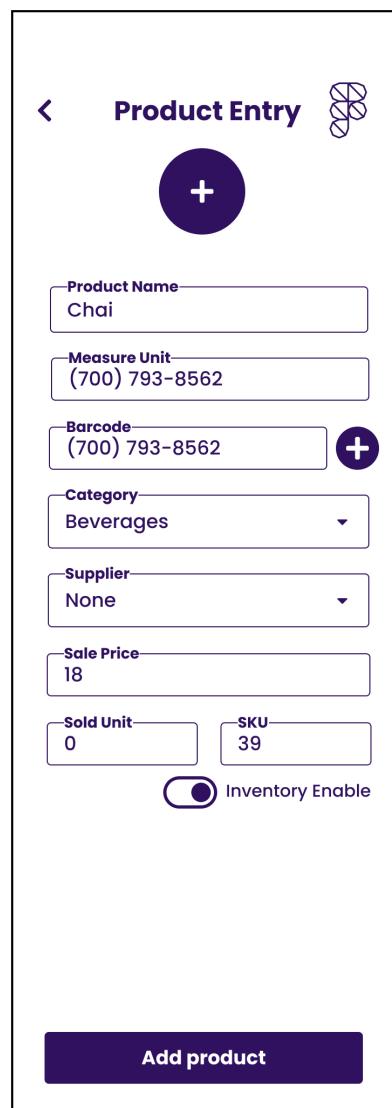


Figure 2.25: Detail Design: Product Entry View

2.3.7 Point of Sales

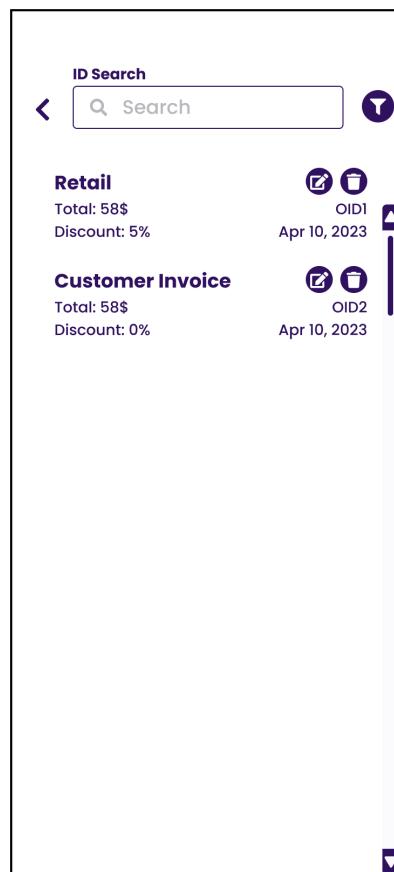


Figure 2.26: Detail Design: Invoice List View

Invoice Entry

Invoice Info:

Invoice ID: OID1

Date: 4/22/2012

Invoice Type: Retail

Customer: None

Items

Product Item

Chai 18\$ 10 boxes x 20 bags	Qty: 1	▼	X
Chang 20\$ 24 - 12 oz bottles	Qty: 1	▼	X

+ Add Item

Billing

Total: 18

Discount: 5%

Note: John Doe

Add Invoice

Figure 2.27: Detail Design: Invoice Entry View

Chapter 3

Implementation

3.1 User Authentication

3.1.1 Login View

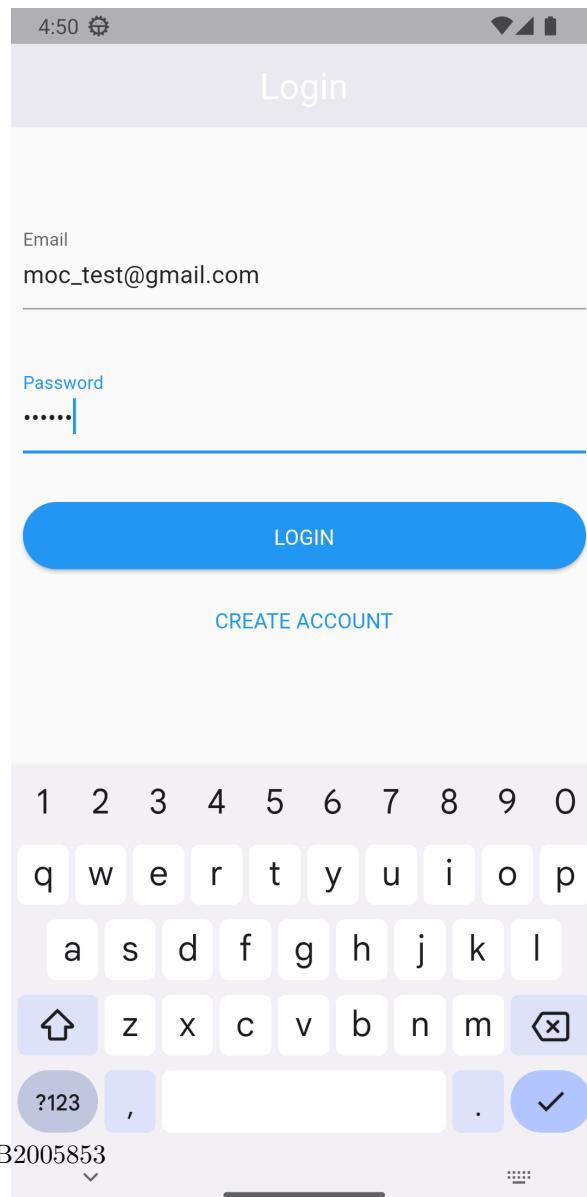


Figure 3.1: Implementation: Login View

3.1.2 Sign Up View

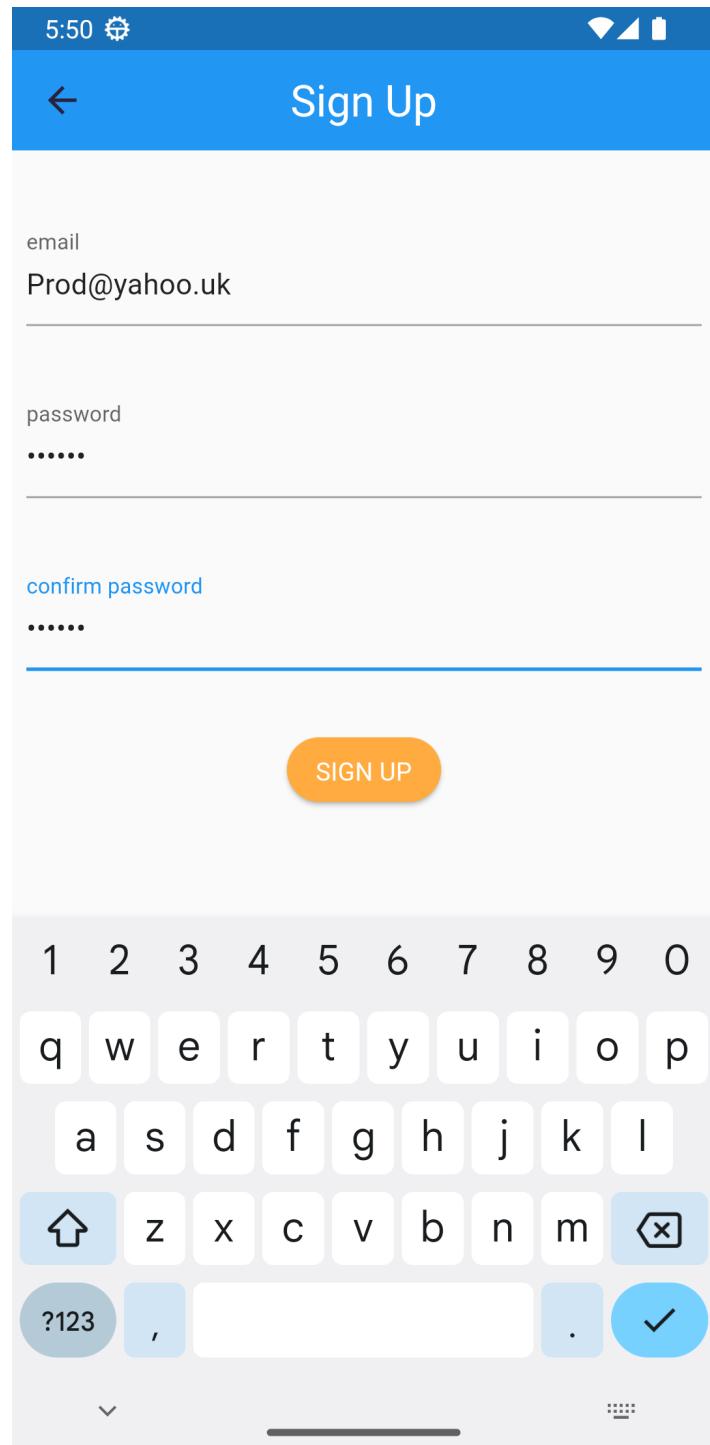


Figure 3.2: Implementation: Sign Up View

3.2 Home View

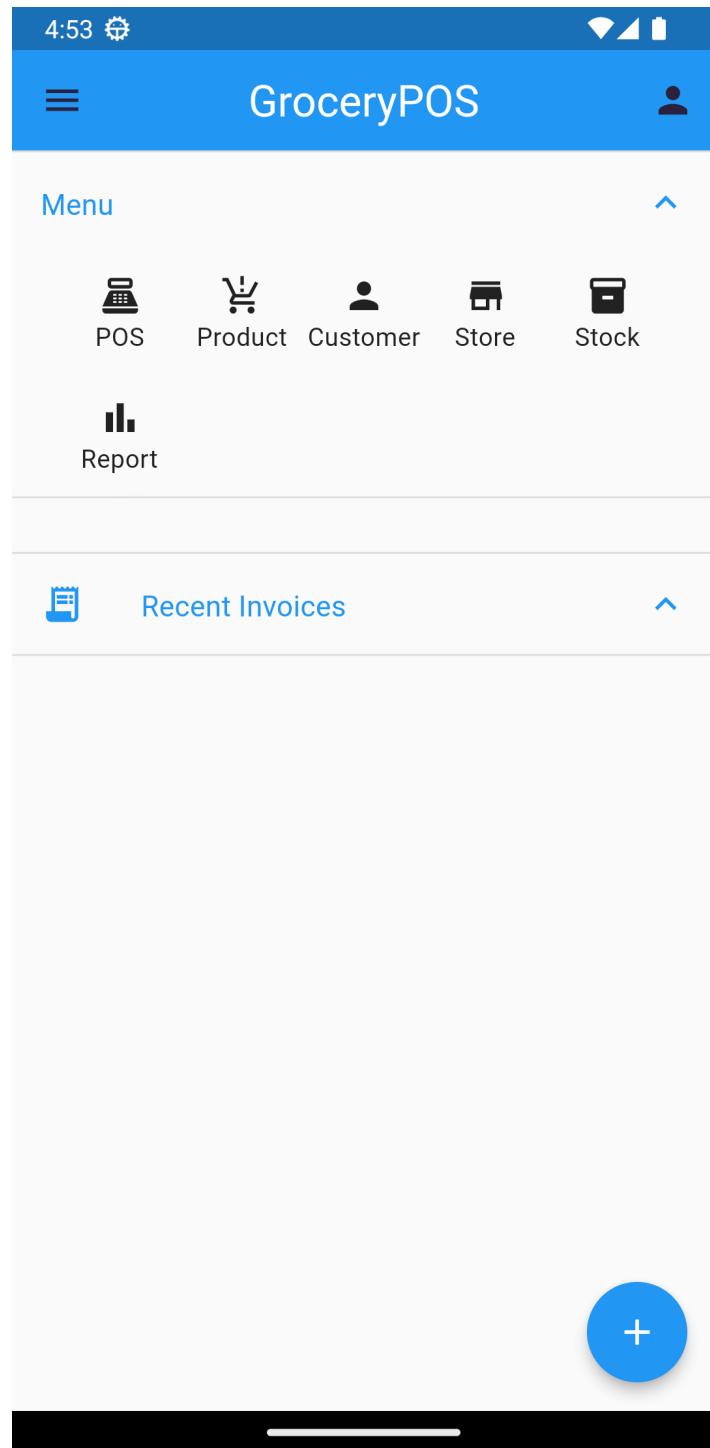


Figure 3.3: Implementation: Home View

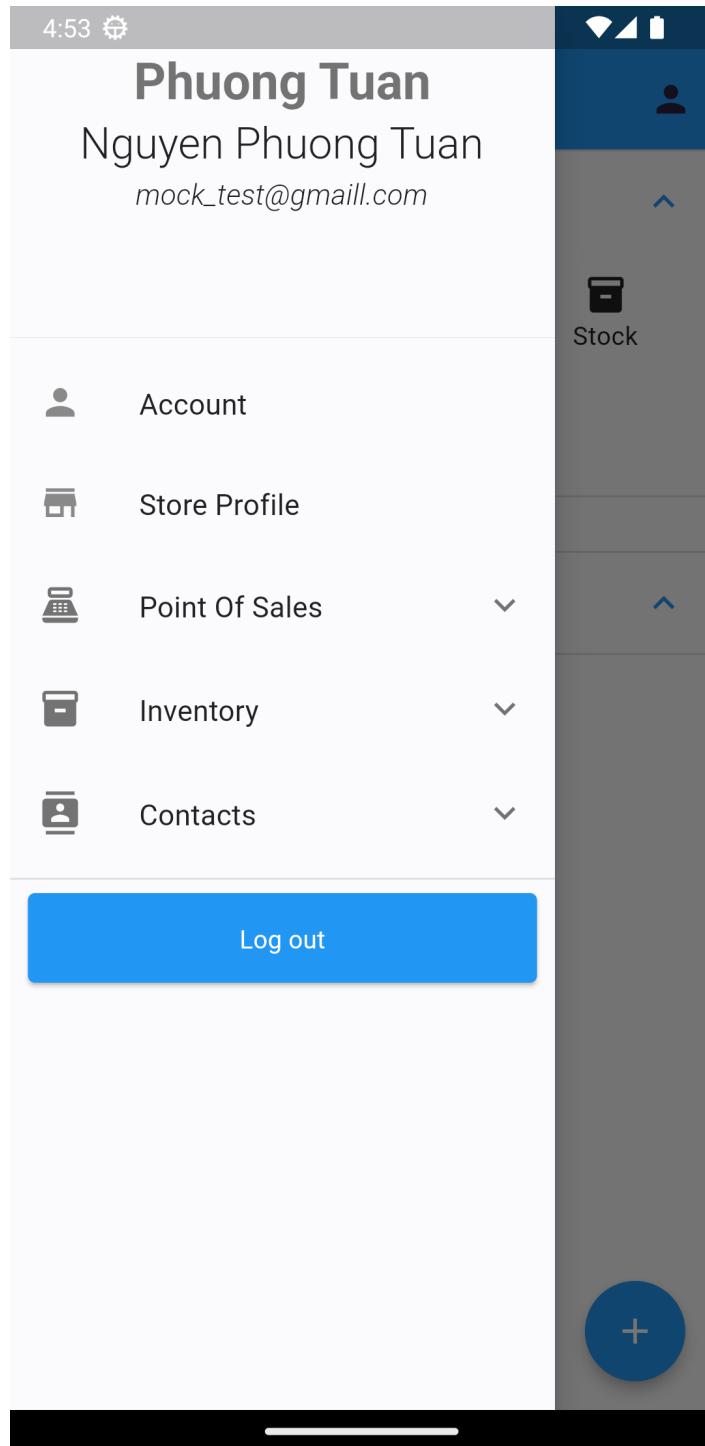


Figure 3.4: Implementation: Home View Side Menu

3.3 User Profile View

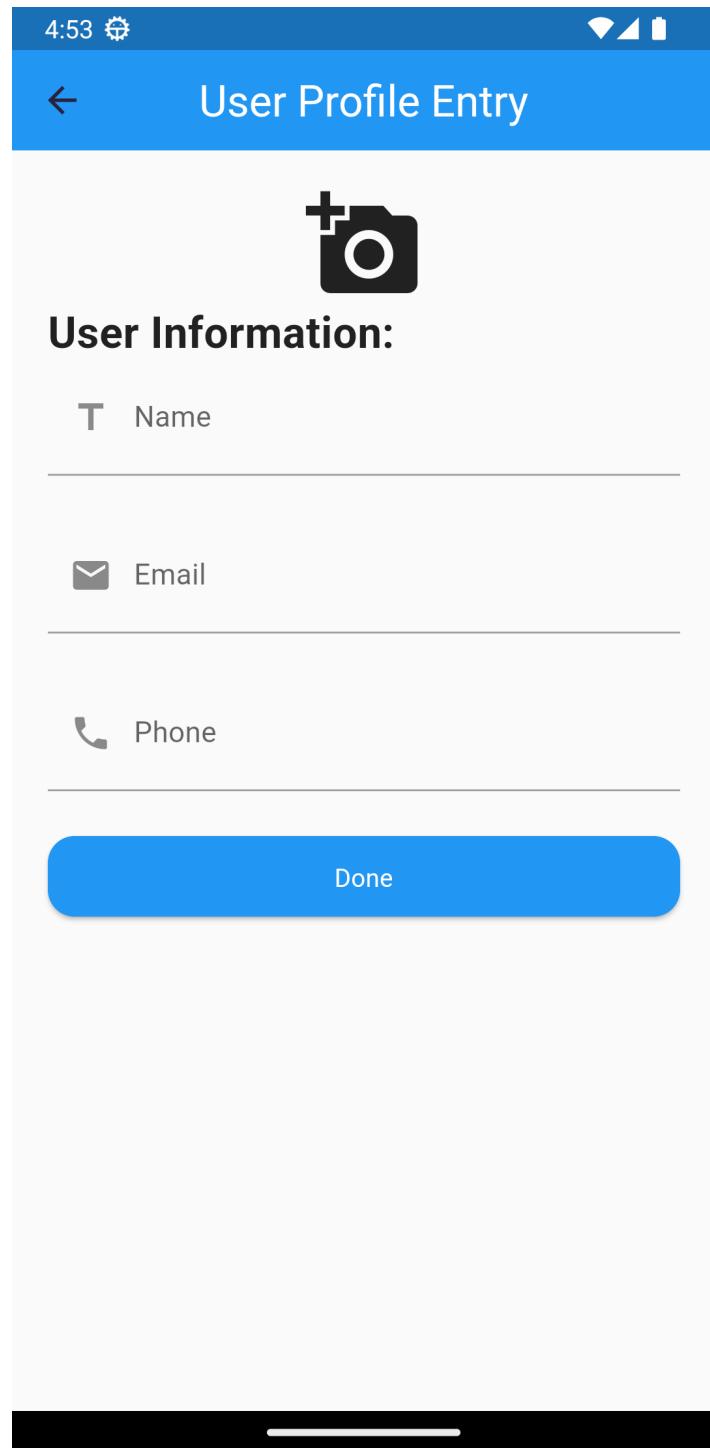


Figure 3.5: Implementation: User Profile View

3.4 Store Profile View

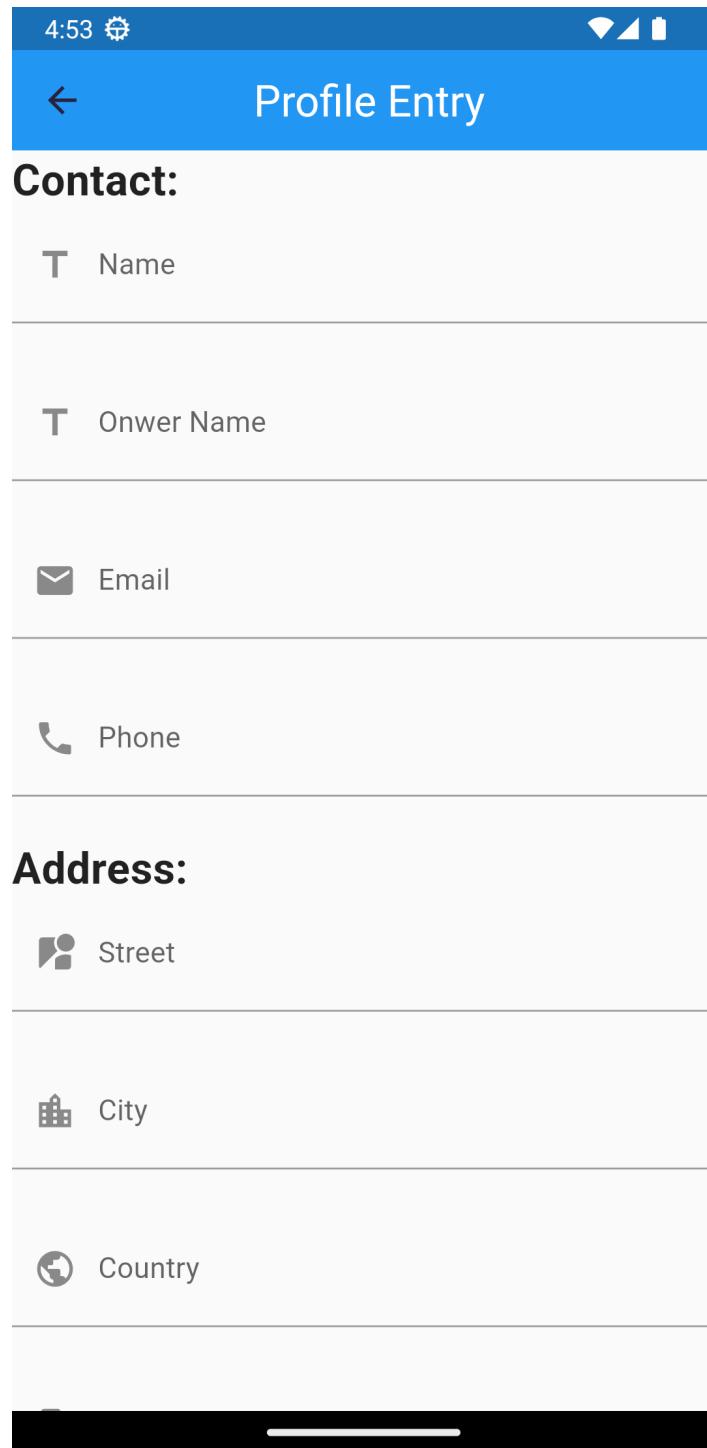


Figure 3.6: Implementation: Store Profile View

3.5 Contacts

3.5.1 Supplier View

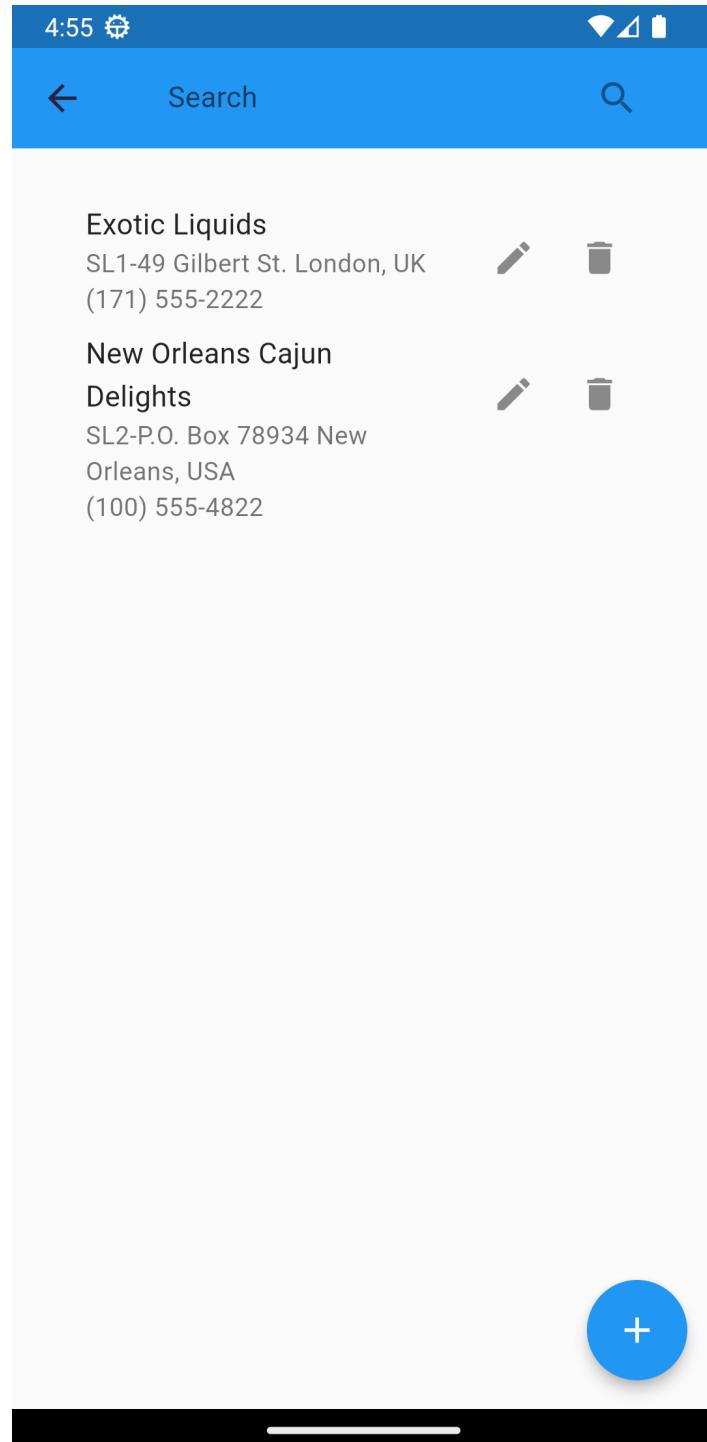


Figure 3.7: Implementation: Supplier List View

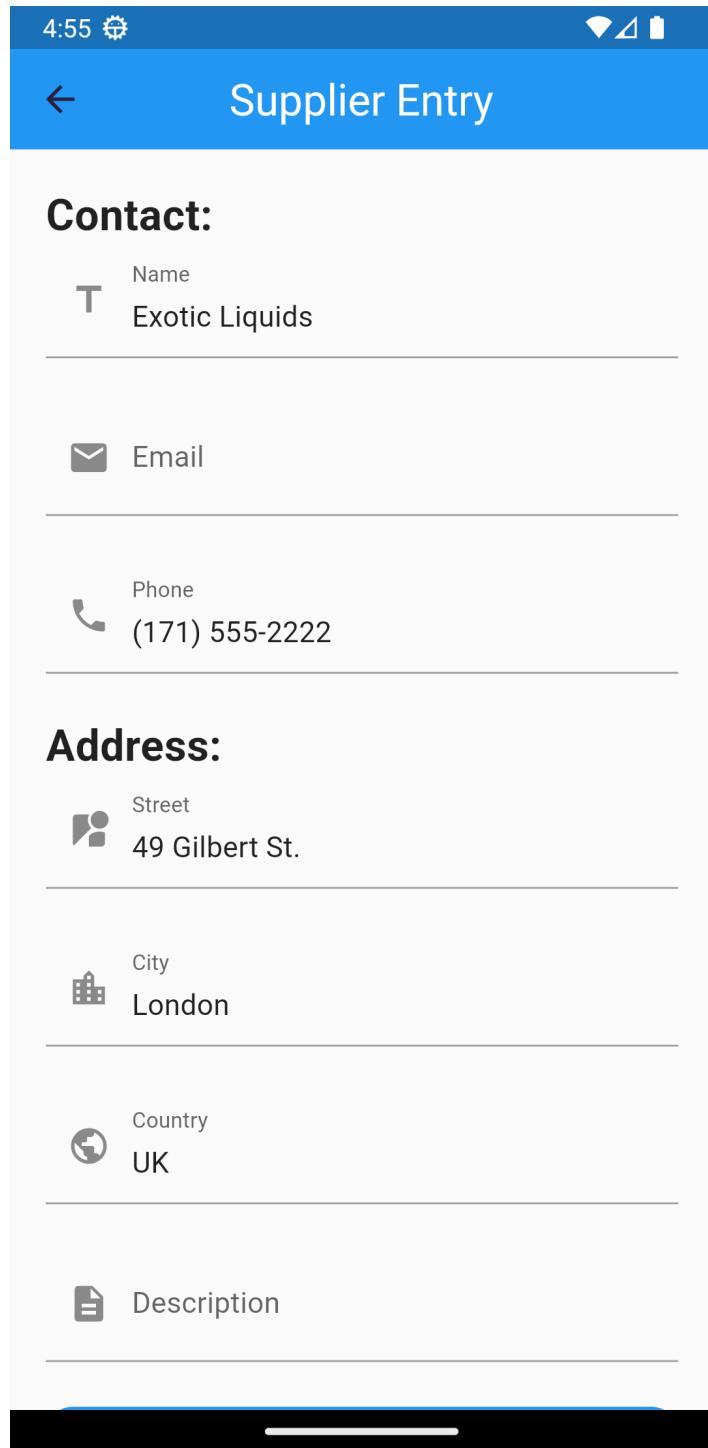


Figure 3.8: Implementation: Supplier Entry View

3.5.2 Customer View

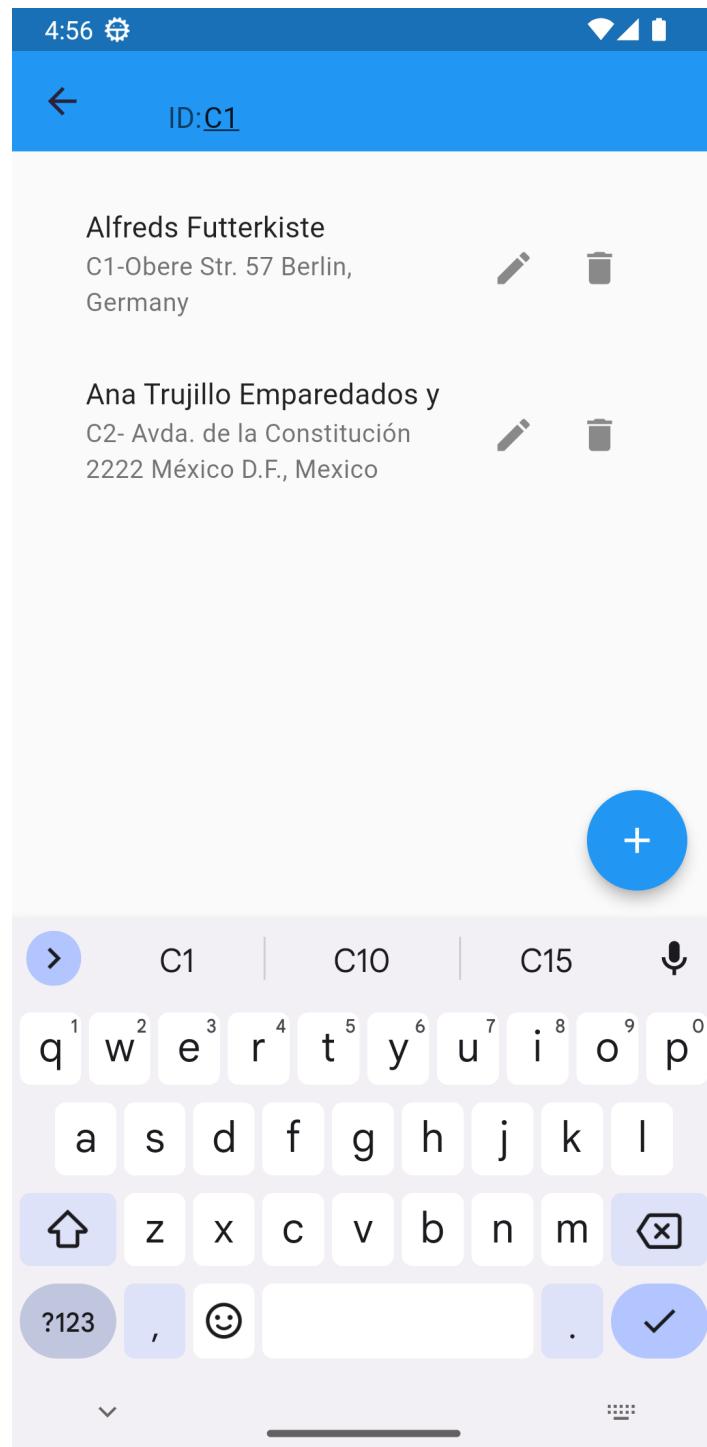


Figure 3.9: Implementation: Customer List View

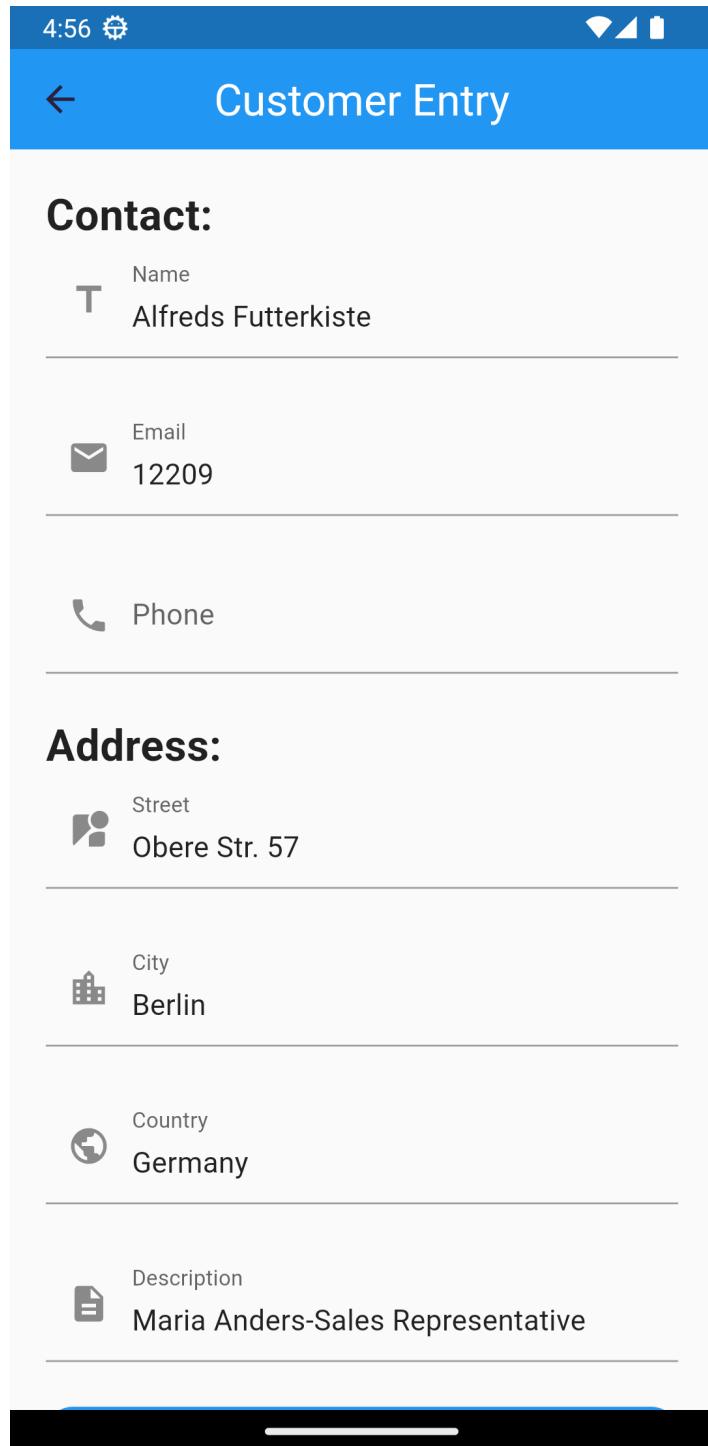


Figure 3.10: Implementation: Customer Entry View

3.6 Inventory View

3.6.1 Category View

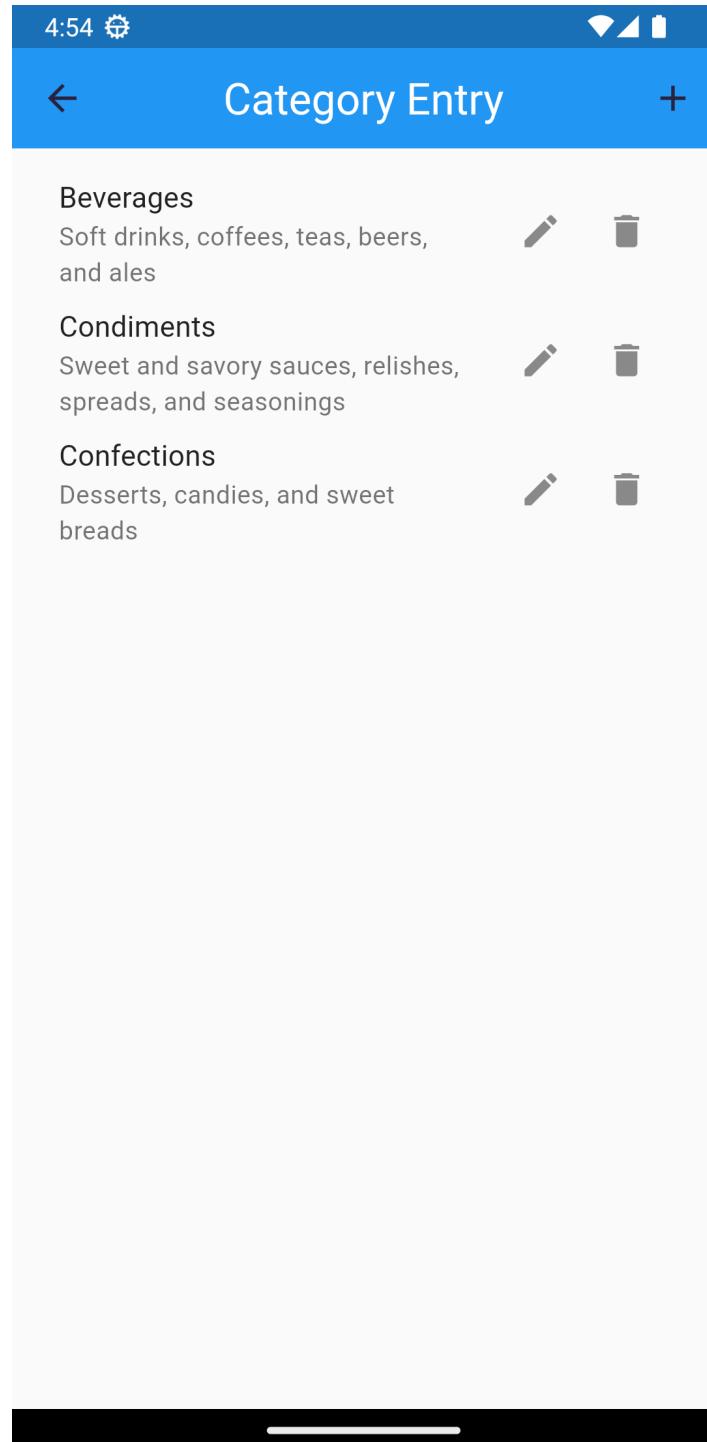


Figure 3.11: Implementation: Category List View

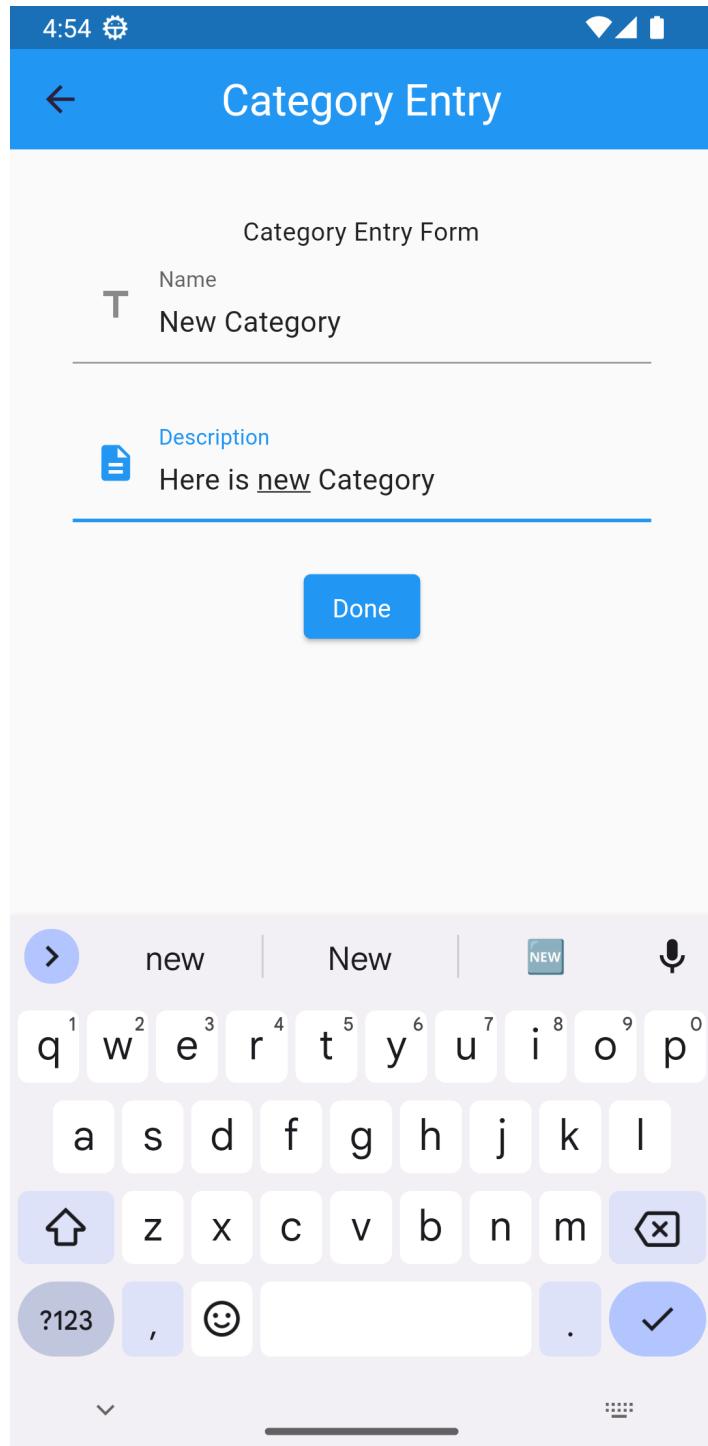


Figure 3.12: Implementation: Category Entry View

3.6.2 Product View

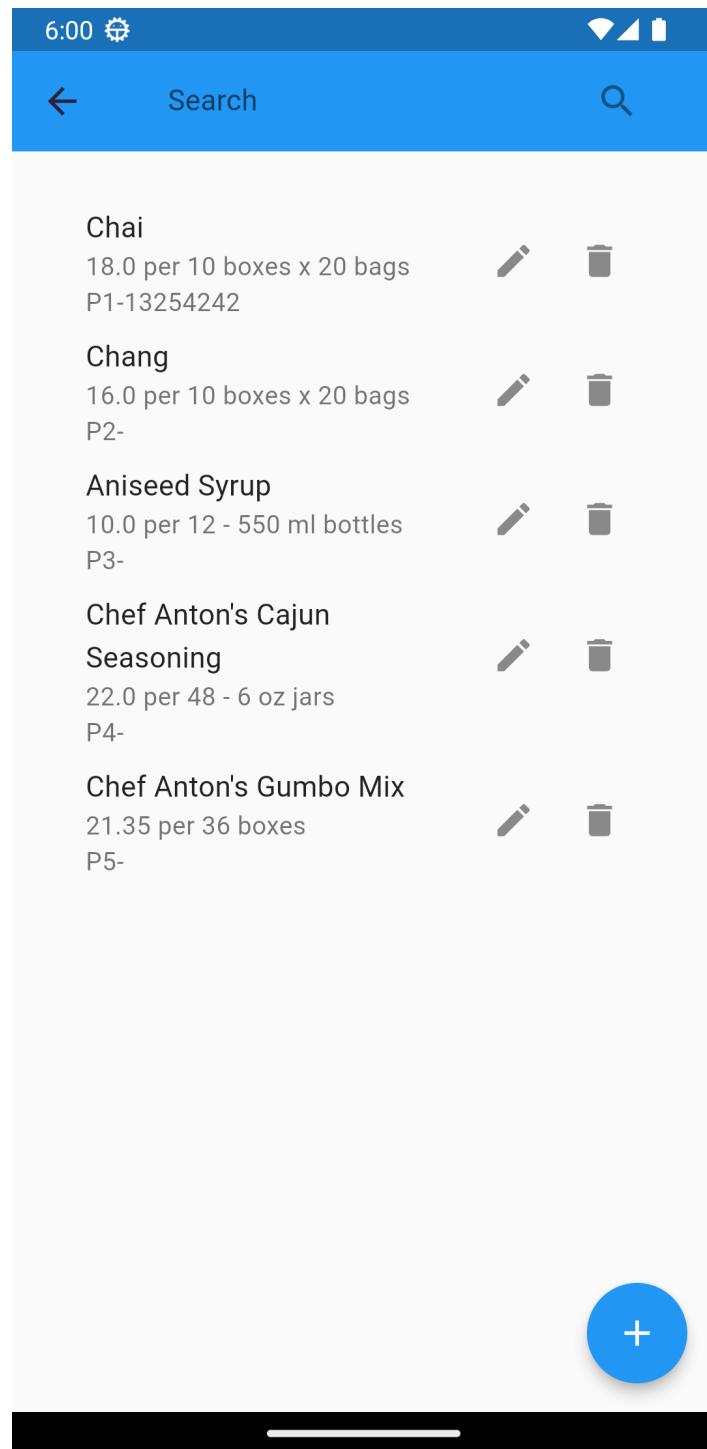


Figure 3.13: Implementation: Product List View

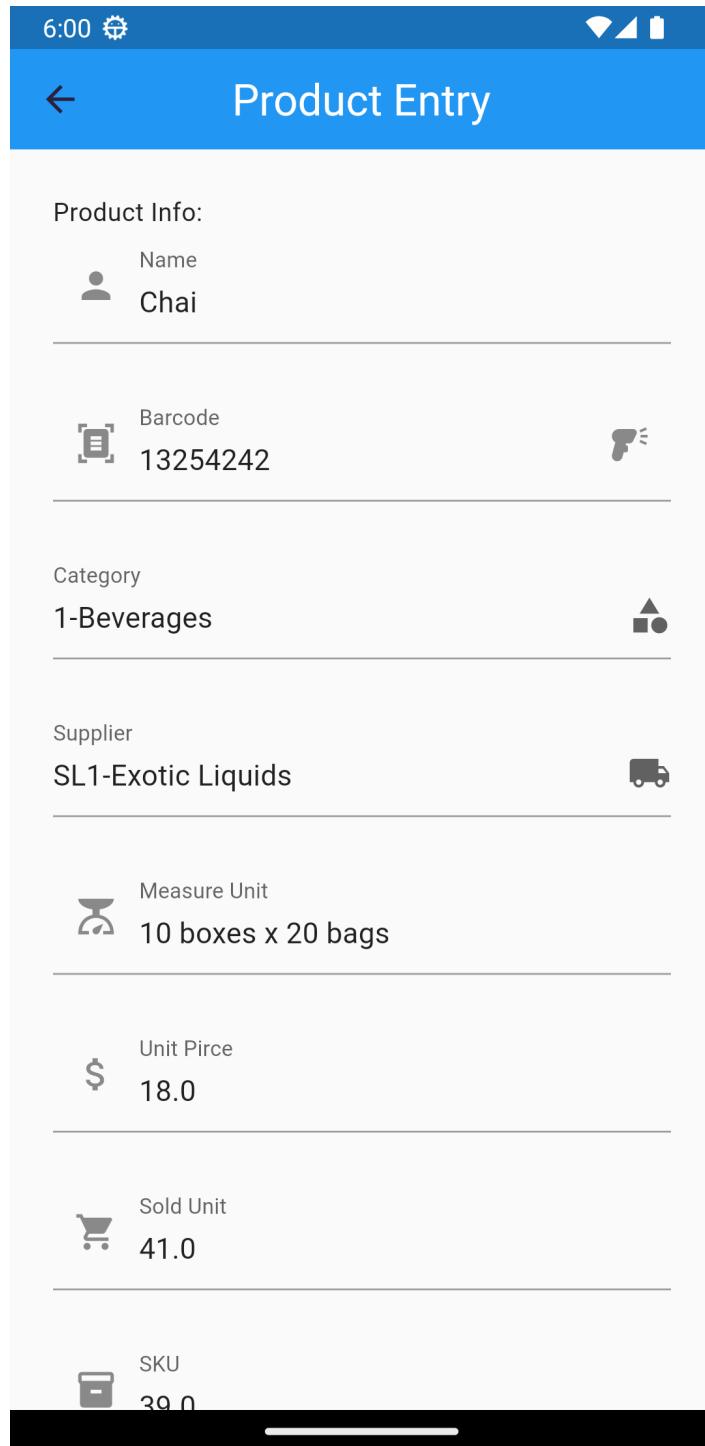


Figure 3.14: Implementation: Product Entry View

3.7 Point of Sales

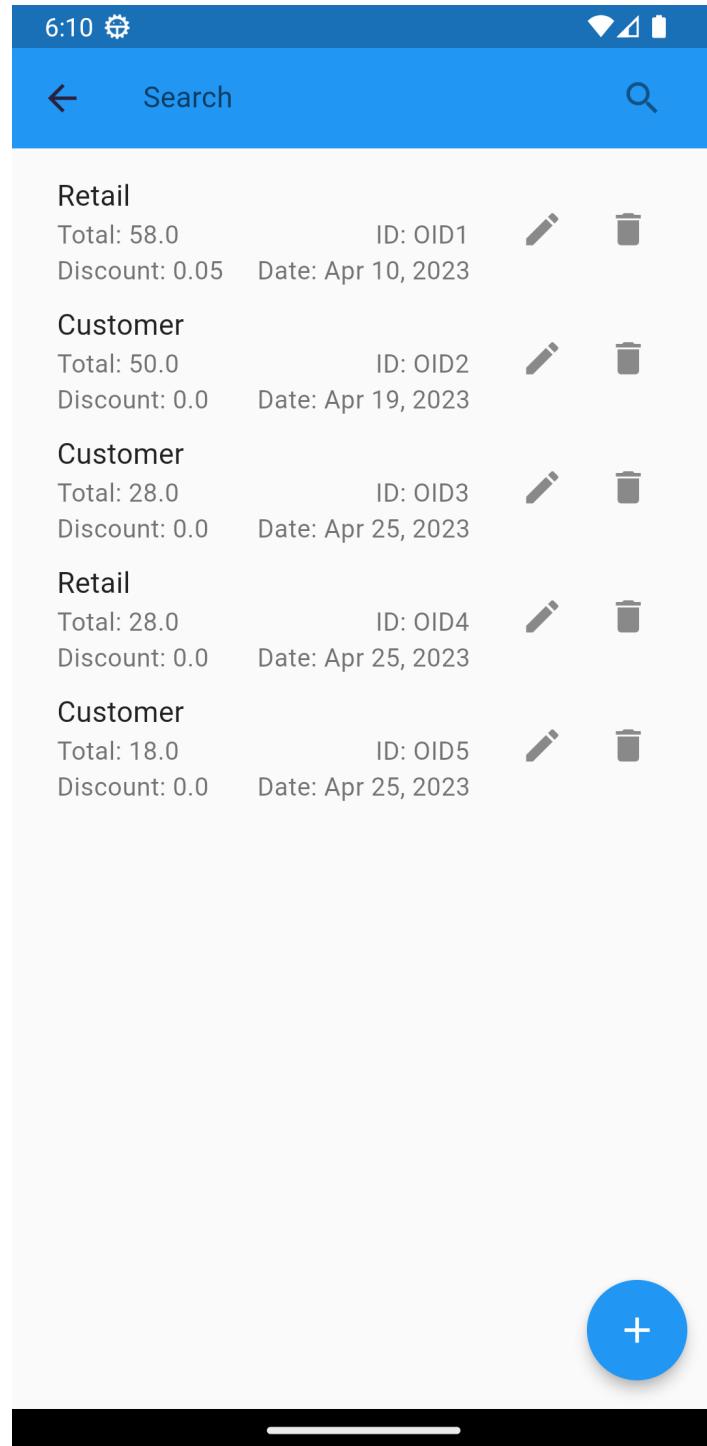


Figure 3.15: Implementation: Invoice List View

Figure 3.16: Implementation: Invoice Entry View - First

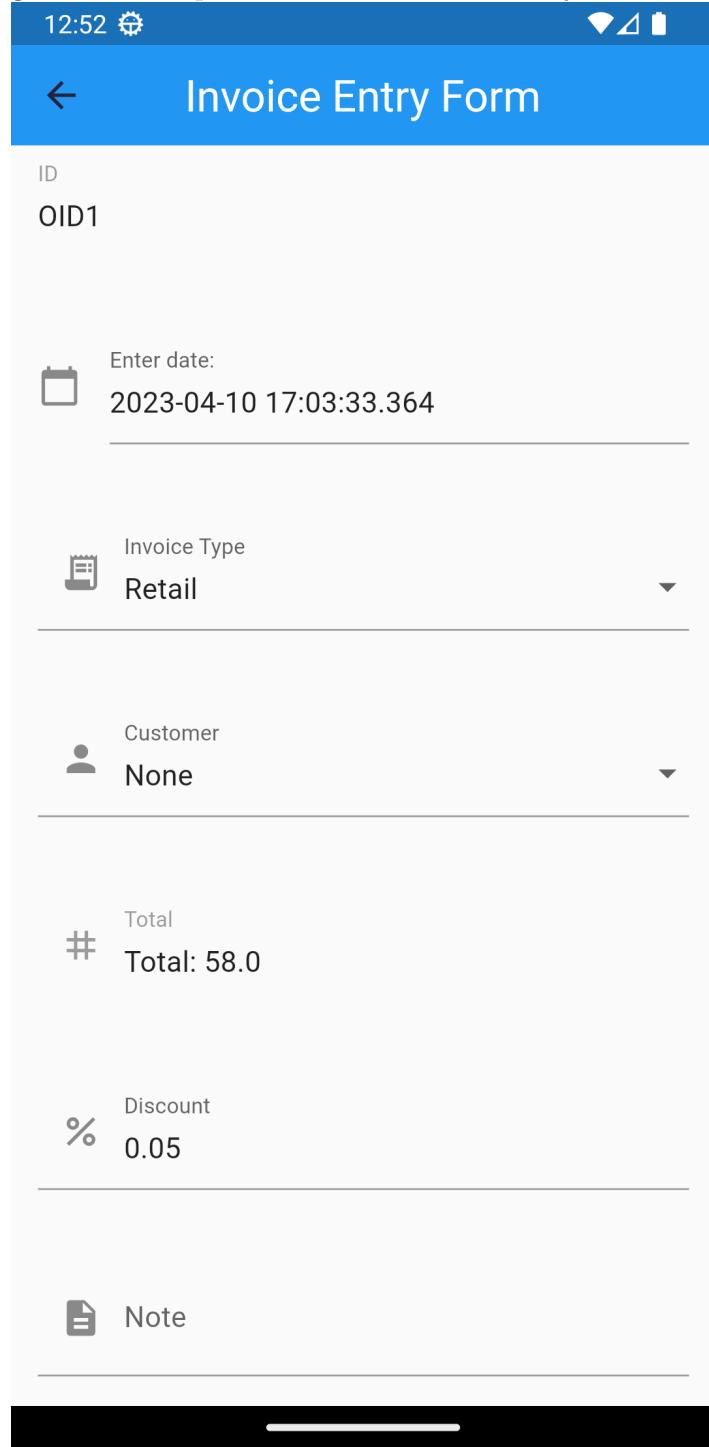
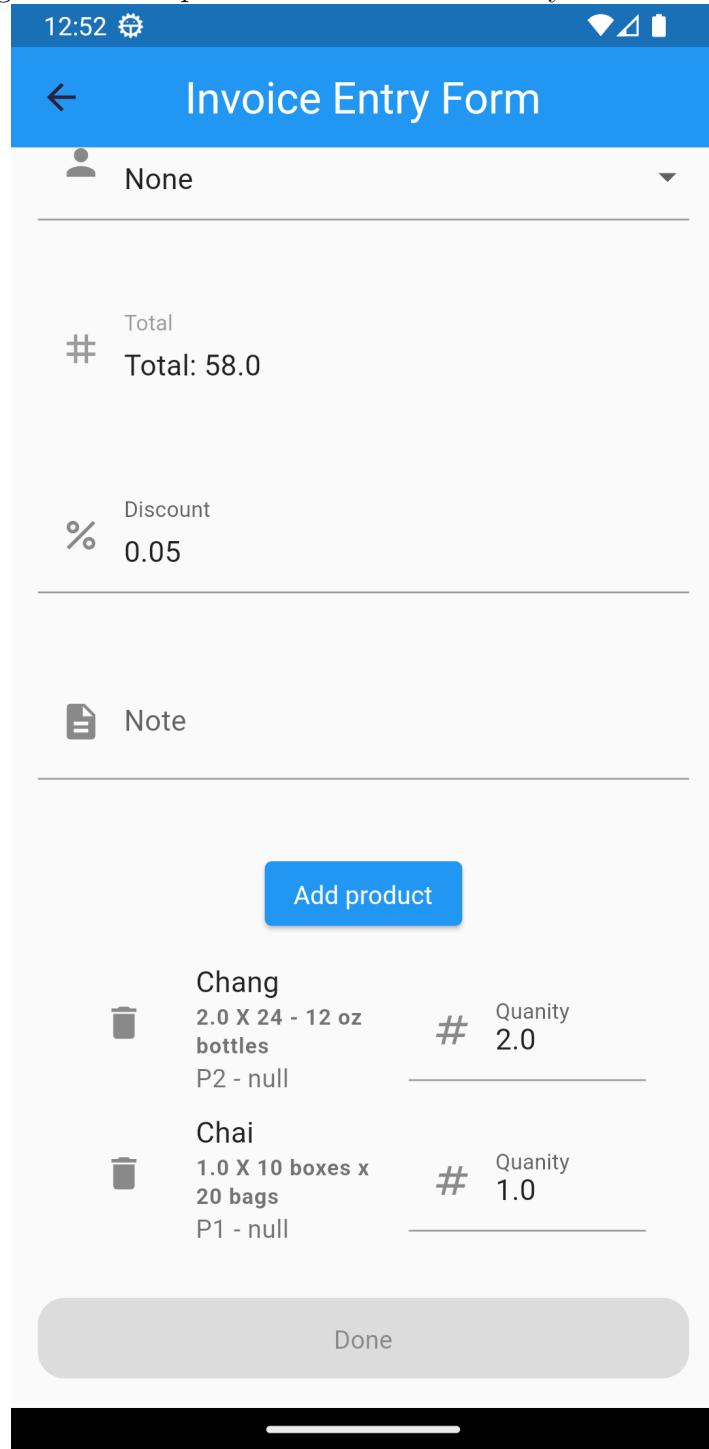


Figure 3.17: Implementation: Invoice Entry View - Second



Chapter 4

Testing

4.1 Introduction

4.1.1 Testing objectives

- Detect errors and verify that the system operates correctly according to the requirements stated in the specification.
- Ensure the completeness of functions.
- Serve as documentation for the maintenance phase.

4.1.2 Testing scope

Create and test some functions in the specification and design documents that meet the expected requirements.

4.2 Evaluation of Testing Results

Evaluation of Testing Results The results of the testing process show that the system achieves high accuracy through multiple tests. The functions are performed according to the original objectives, and data queries are accurate.

Part III

Conclusion

Achievements

Achievements After researching and implementing the project, I have completed the development and evaluation of a mobile application solution for small businesses to assist in managing sales and inventory. The application was developed using the Flutter framework and utilized Firebase as the data platform. The specific objectives of the research were to:

- Investigate challenges faced by small businesses in managing sales and inventory.
- Develop a mobile application solution for small businesses using the Flutter framework and Firebase.
- Evaluate the usability and effectiveness of the developed application through user testing and surveys.

The outcomes of this study will contribute to the knowledge of mobile application solutions for small businesses.

Limitations

- The interface is still simple
- Self-researched data is not entirely correct
- Not all functions have been processed for each form.
- Only search by ID is correct because Firebase does not support full-text search
- Although the data has image URLs, images cannot be loaded from Google Cloud.

Development Direction

Due to limited knowledge, it is necessary to learn and use retail business processing more effectively. In addition, some other functional modules need to be added for better software operation such as adding invoice and type constraints, adding data flow processing as well as better error handling from Firebase.