Mobile POS System

16/ENG/016

BRBSB Baruwaththa

Cover Letter

BRBSB Baruwaththa,
16/ENG/016
Faculty of Engineering,
University of Sri Jayawardenepura,
Rathmalana

24/11/2023

Dr Randima Dinalankara,
Senior Lecturer,
Faculty of Engineering,
University of Sri Jayawardenepura,
Rathmalana

Dear Sir,

Submitting Final Report of Computer Engineering Project 'Mobile POS System'

I am pleased to submit the final report of the 'Mobile POS System' which I am developing for the CO3302 Computer Engineering Project course.

This is a small-scale product for low cost, easy to handle, alternative option to current complex POS system.

Thank you.

Sincerely,

BRBSB Baruwaththa – 16/ENG/016

Contents

Introduction

Title: Mobile POS System

The Mobile POS system is an android based POS app that less depend on other hardware components. The camera feature of mobile phone is used as the bar code reader and it is very easy to handle and contain with easy understanding mobile UI. It can scale up with enough implementations to use large scale business environment. The system tracks all transactions, stock changings and generate necessary details. Specially Sri Lanka faces high inflation, then traders face to unnecessary risk and competition of price changing. Here Cost averaging algorithm added to calculate averaged buying price and selling price for get decisions for shop owner.

Objectives

- Introduce a low-cost POS system alternative for the current popular systems.
- Ability to optimize the possibility of using mobile devices for business processes as a multi-purpose device.
- Simplify the pos operations with mobile UIs and its functionalities.
- Reduce the usage of hardware for better cost management and indirectly affect to save environment

Architecture

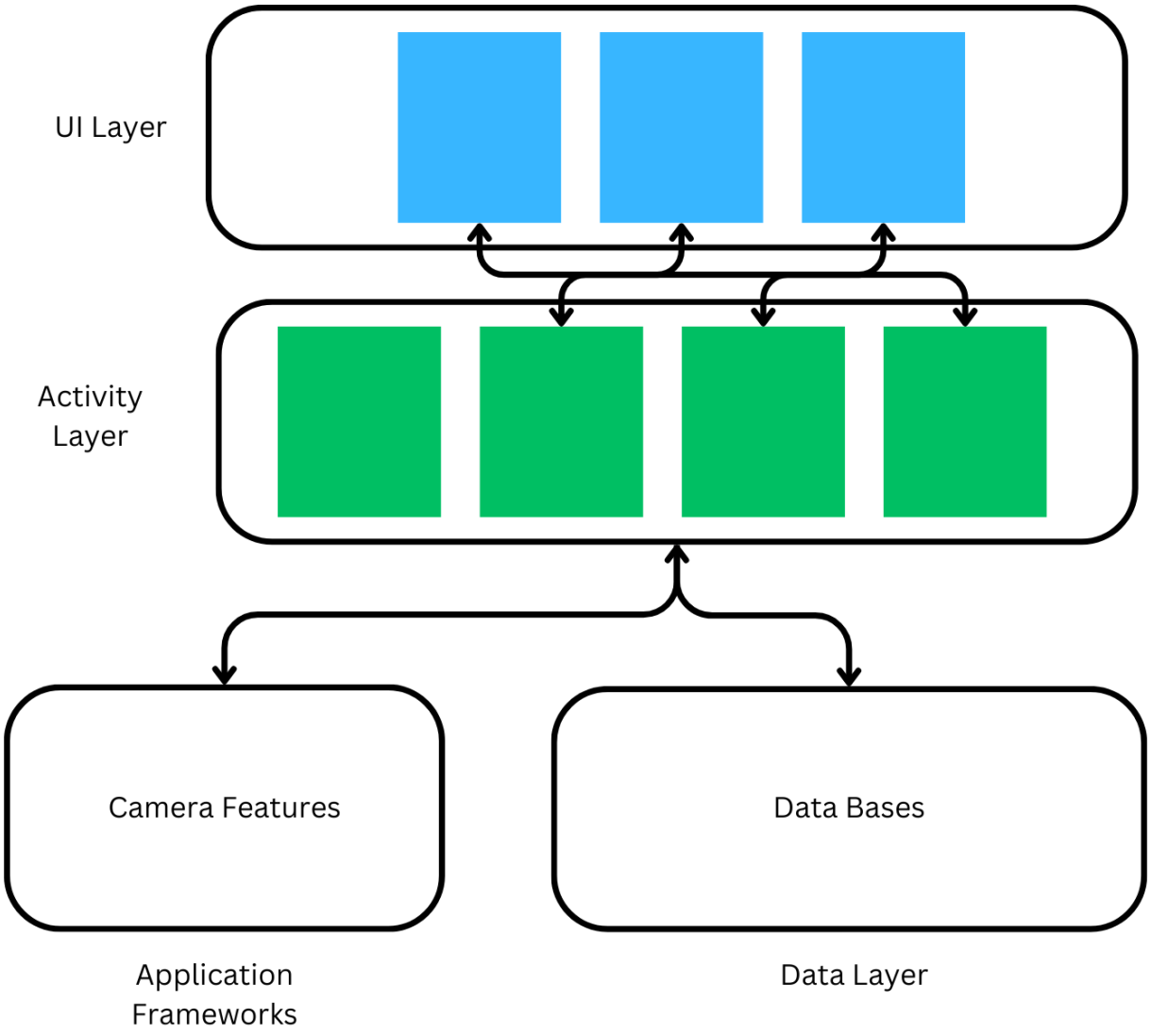


Image 1: Use Case Diagram

UI Layer: User interacts through Layouts. There are layouts for each page and template layouts for components in it.

Activity Layer: Activities belongs to layouts of UI. It handles the operations, navigations through app pages, collaborate with OS, hardware and other features.

Data layer: contained the stores data in resources such as databases. Repositories are connect activities and databases

Application Framework: connect hardware and other OS features with application. (Camera, notification, broadcasts, media)

Design

There are 3 UML diagrams are included to this project report.

(1) Use Case Diagram

(2) Class Diagram

(3) Activity Diagram

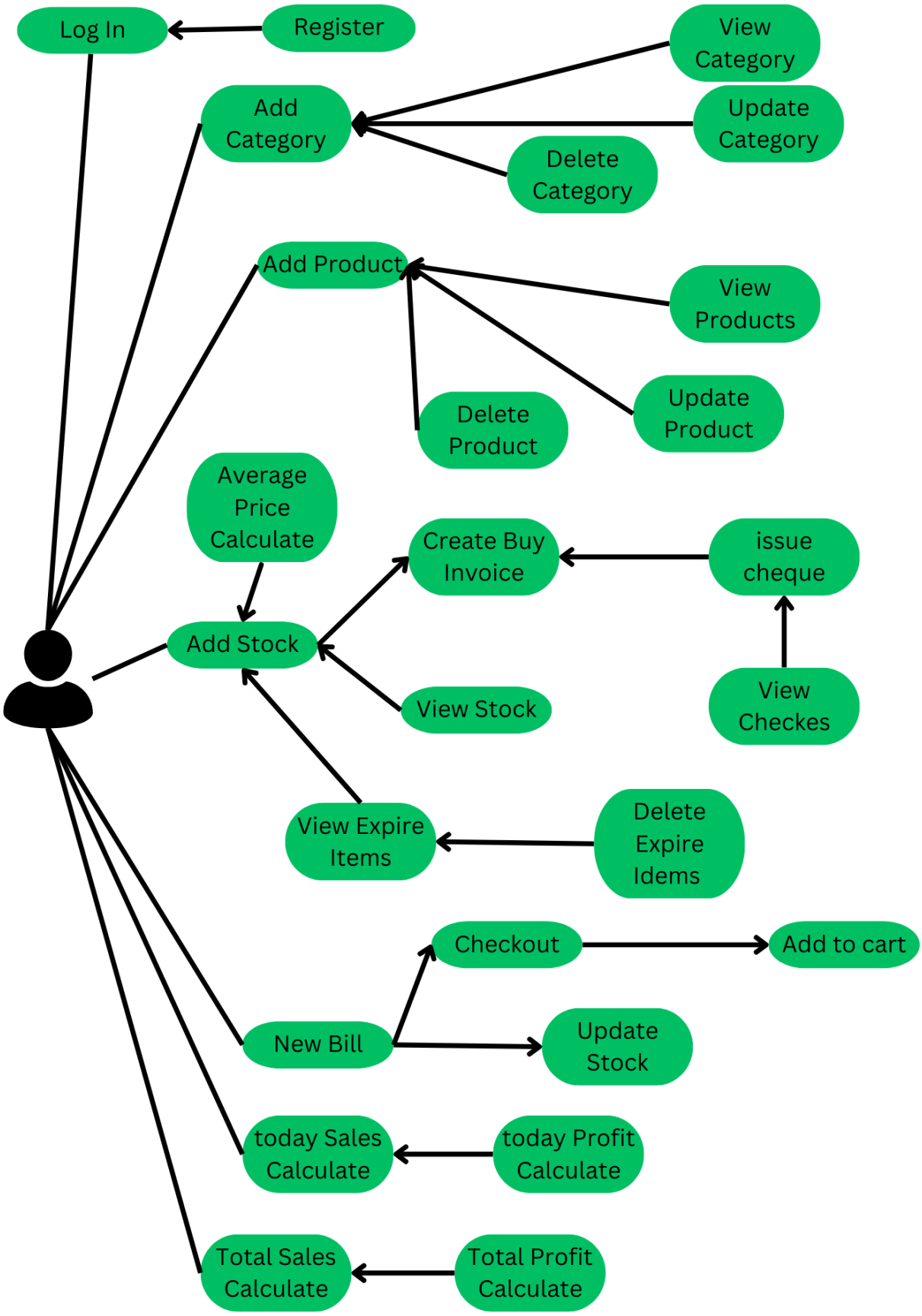


Image 2: Use Case Diagram

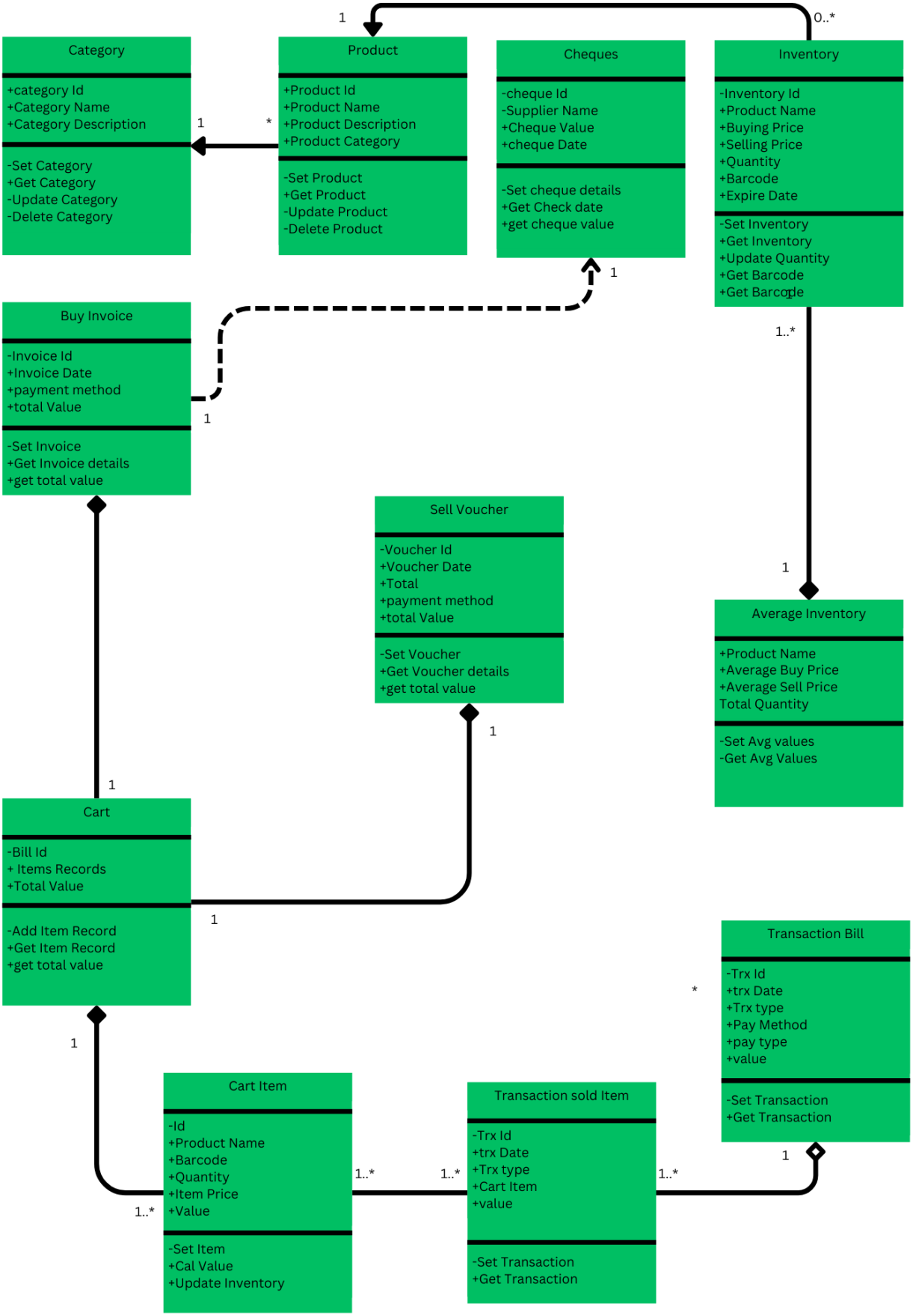


Image 3: Use Case Diagram

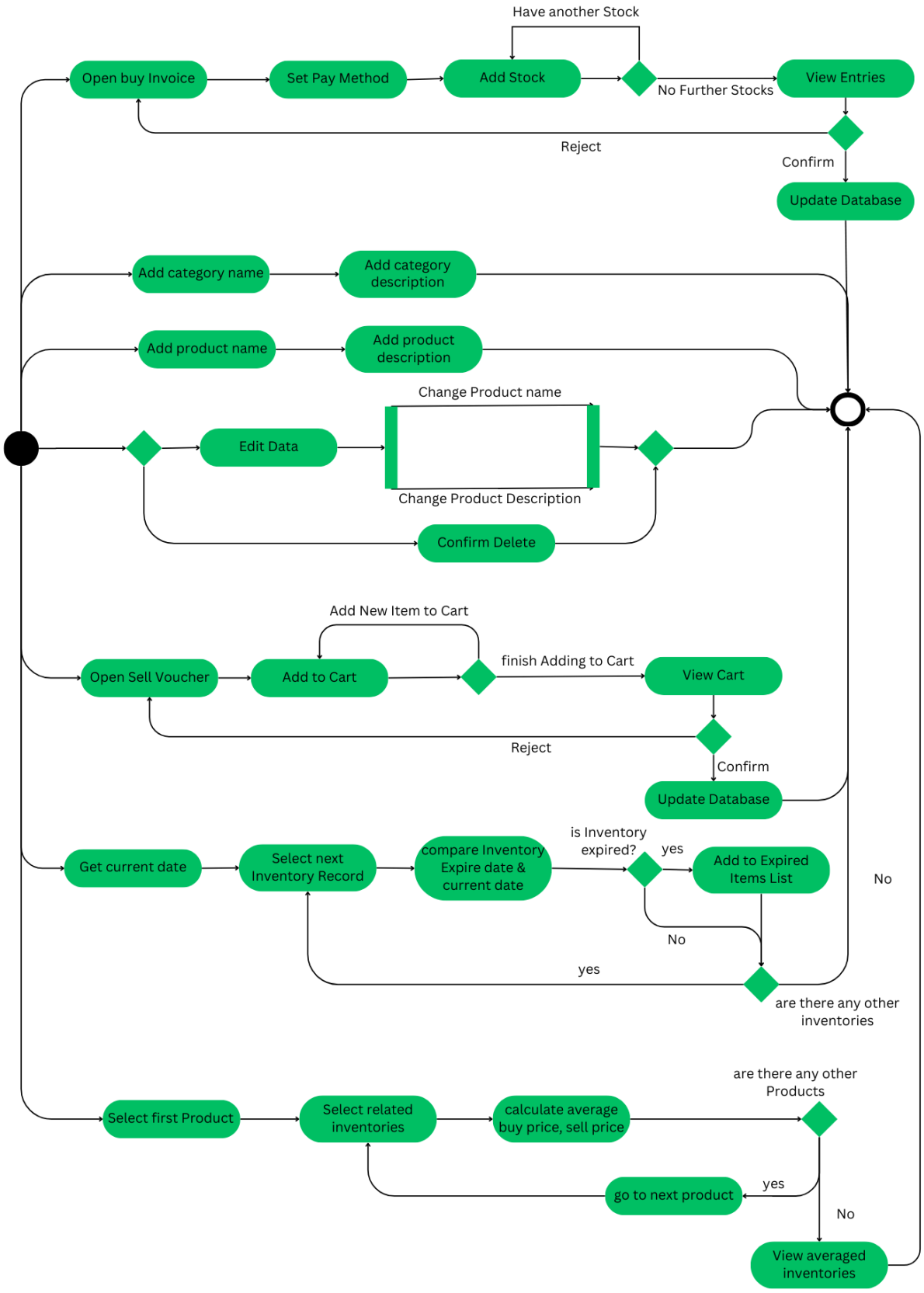


Image 4: Use Case Diagram

Technologies

Android Studio

Android studio is a special version of Intelli-J software for android app development.

XML

XML is used for developed UIs for android apps. Google material design library is used to shape the UI components.

JAVA

Java is the main language use to run the back-end.

SQLite

SQLite is used for do CRUD operations about databases.

Hardware

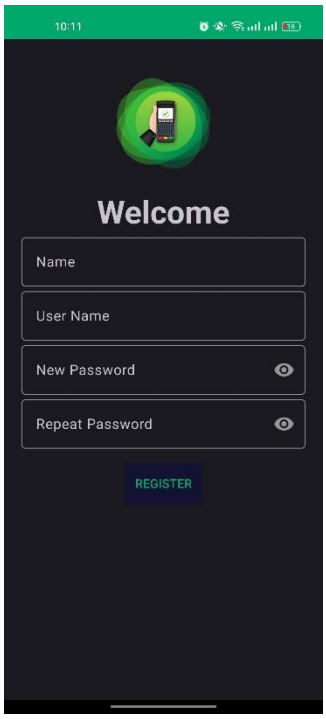
2 Mobile phones (Oppo & Samsung) and Tablet (Samsung S6 lite).

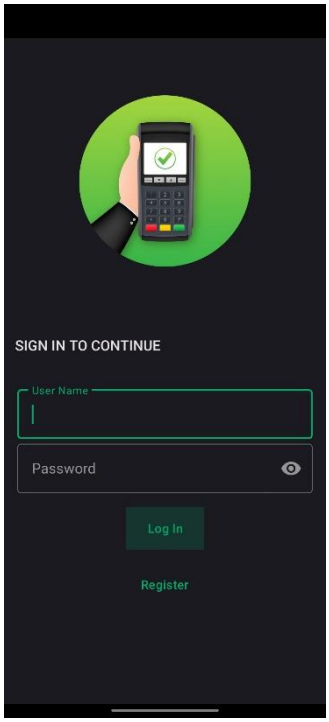
Milestone & Completion

	Requirement Gathering	Develop the system architecture	Developing the pages		Test the integration	Finalize UI & error handling
			Layout Developments	Activities Development & Unit testing		
Sep-04						
Sep-11						
Sep-18						
Sep-25						
Oct-02						
Oct-09						
Oct-16						
Oct-23						
Oct-30						
Nov-06						
Nov-13						

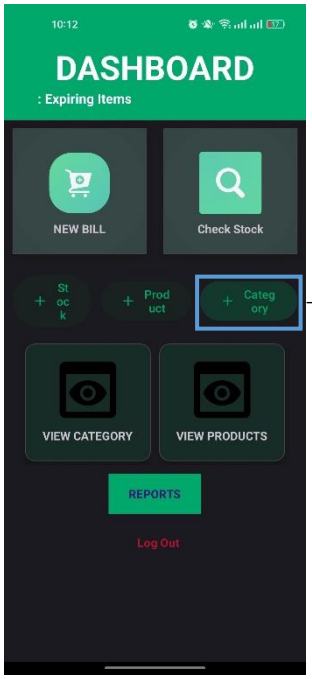
GUI with Guidance

Register & Login

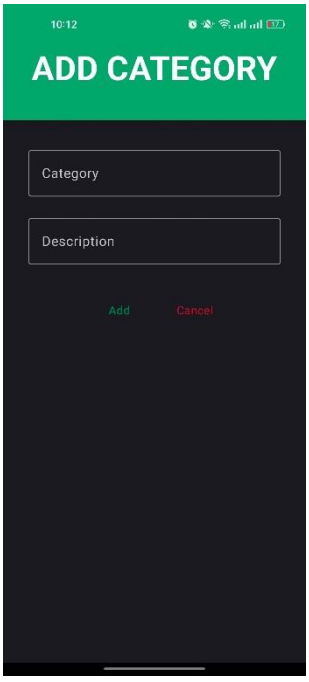




Add Category

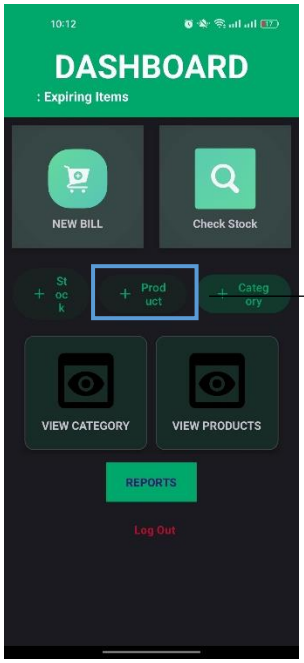


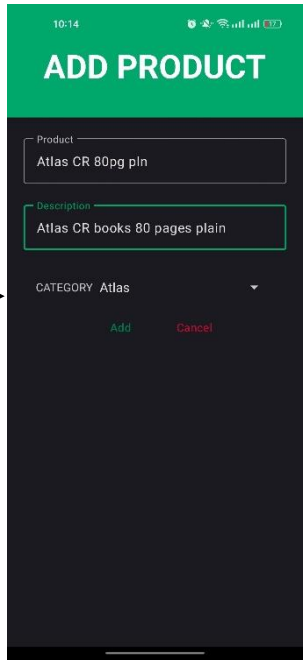
Click "+ Category"



Fill details and click

Add Products





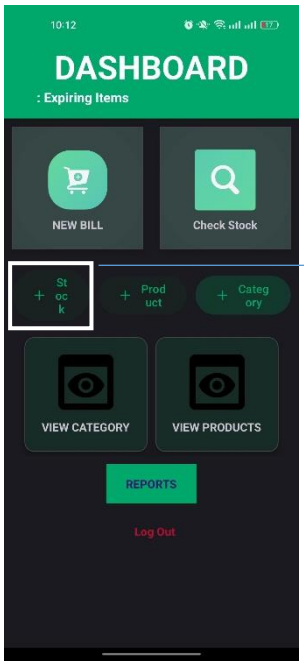
Click “+ Products”

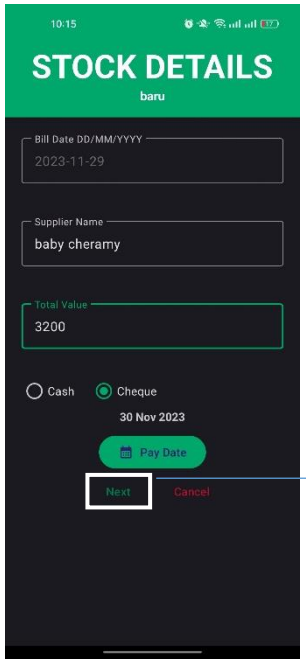
Fill details

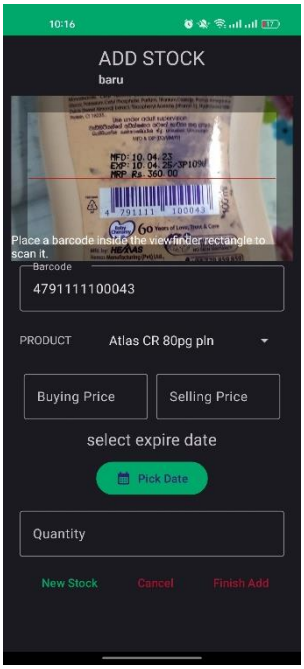
Select Category

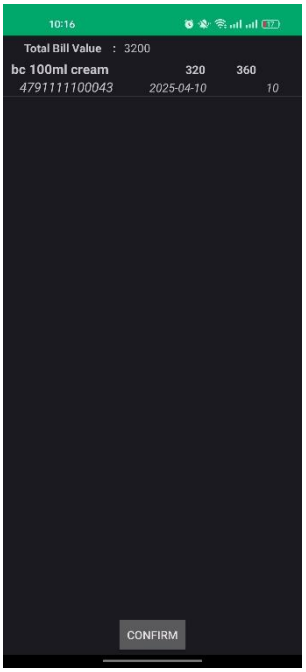
Click “Add”

Add Stock to Inventory









Click “+ Stock”

Fill Details

Select Pay method

Select date if cheque selected

Scan Barcode

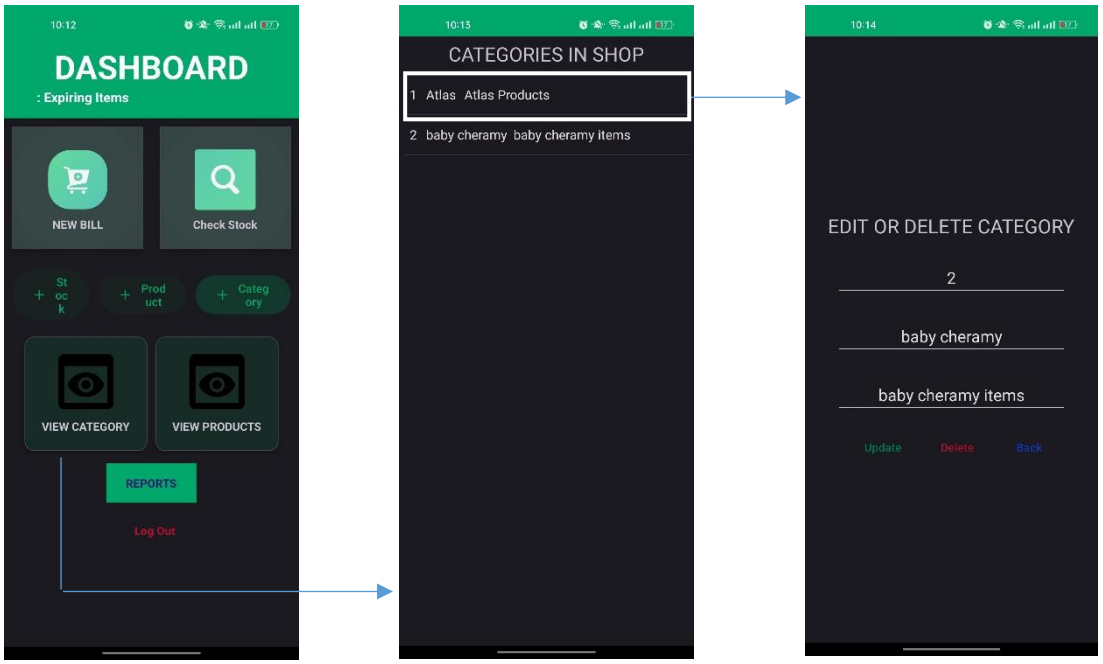
Select Product

Set “Buy Price”, “Sell Price”, Expire Date and Quantity

Click “New Stock” to add more

Or click finish to next

View Categories & Update or Delete Categories

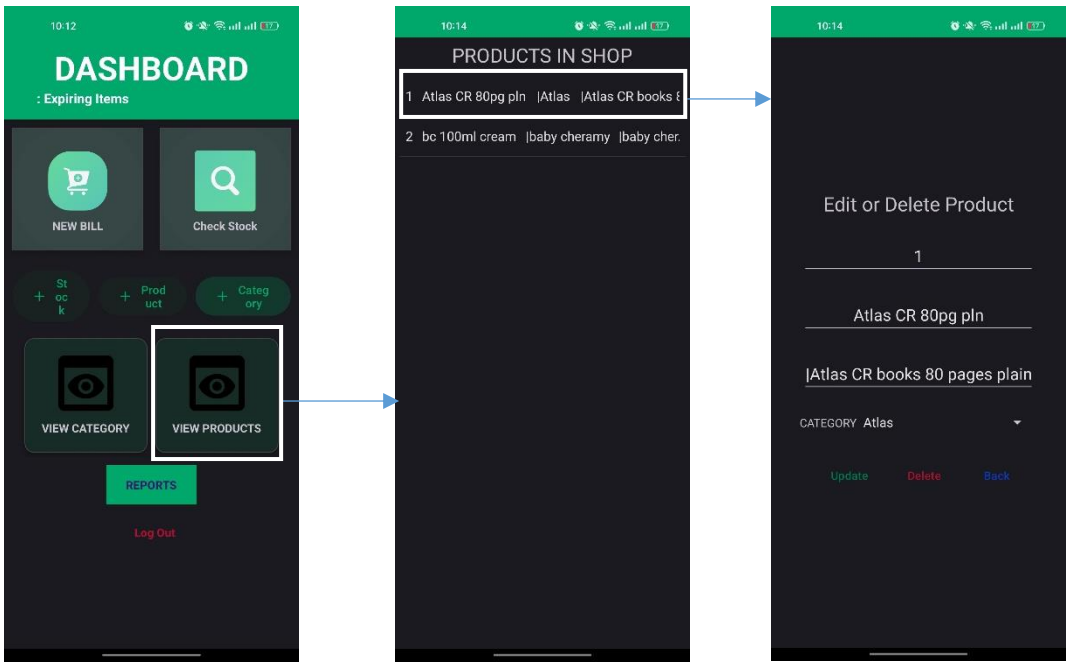


Click “View Category”

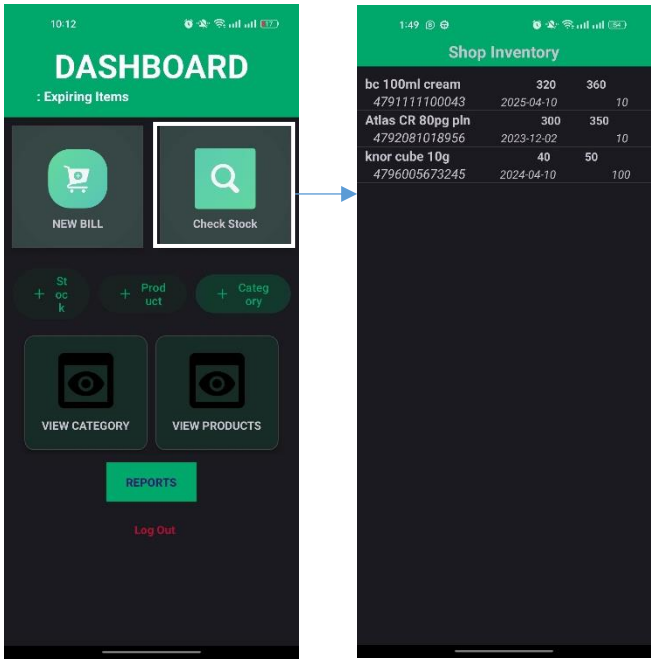
Click a row

Edit details and click update or delete

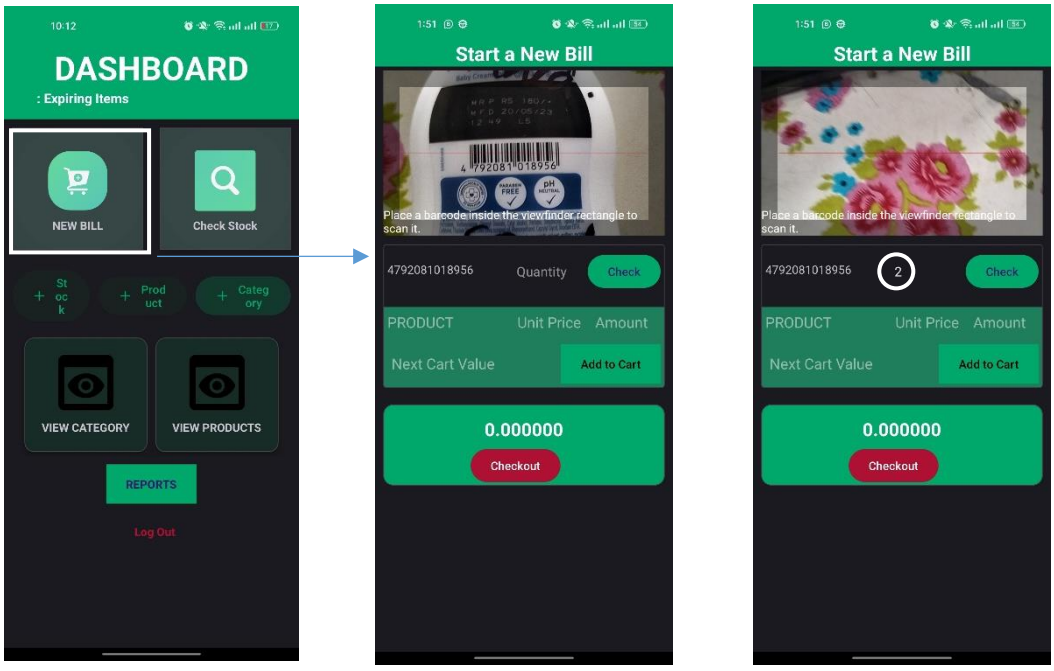
View Products & Update or Delete Products



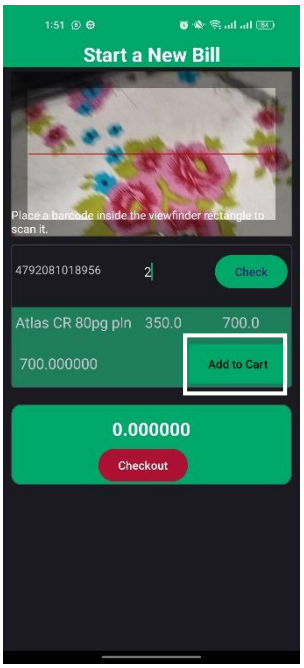
View Inventory



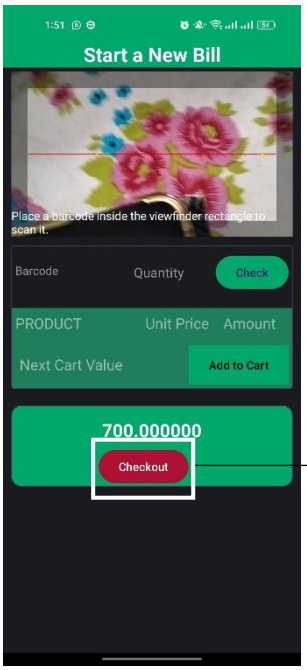
Sell Items



Click “NEW BILL” Scan Barcode Set quantity and click “Check”



Click “Add to Cart”

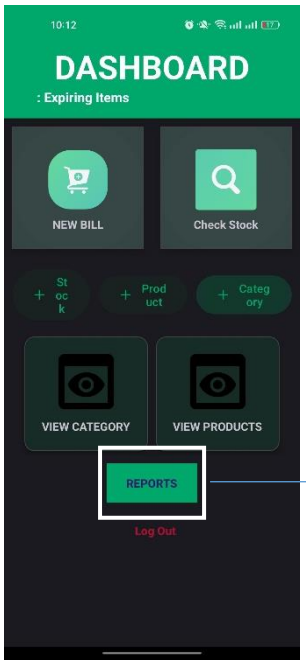


To add new Item to cart: repeat process

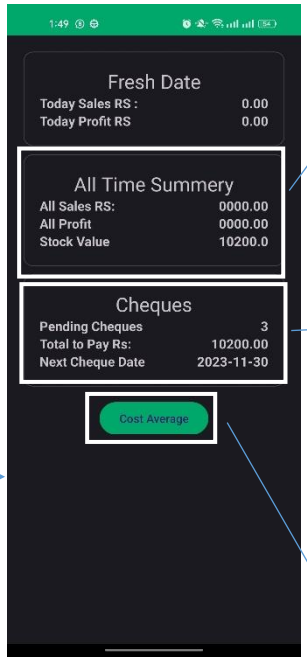
Or Click Checkout to next

Click “Confirm”

Reports



Click “Reports”



The Activity present

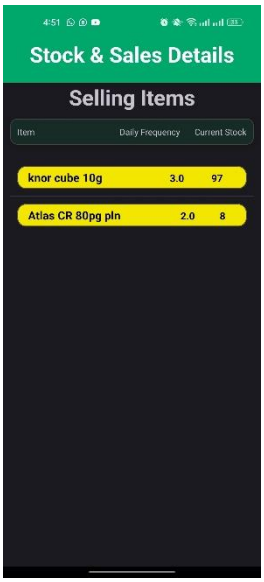
: Current Date Sales Details

: Total Operation Details

: Cheques Details

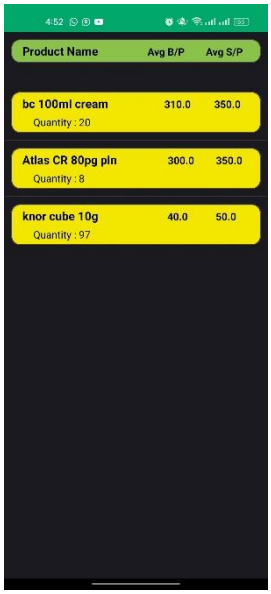
If need calculate cost average,

Click “cost average”



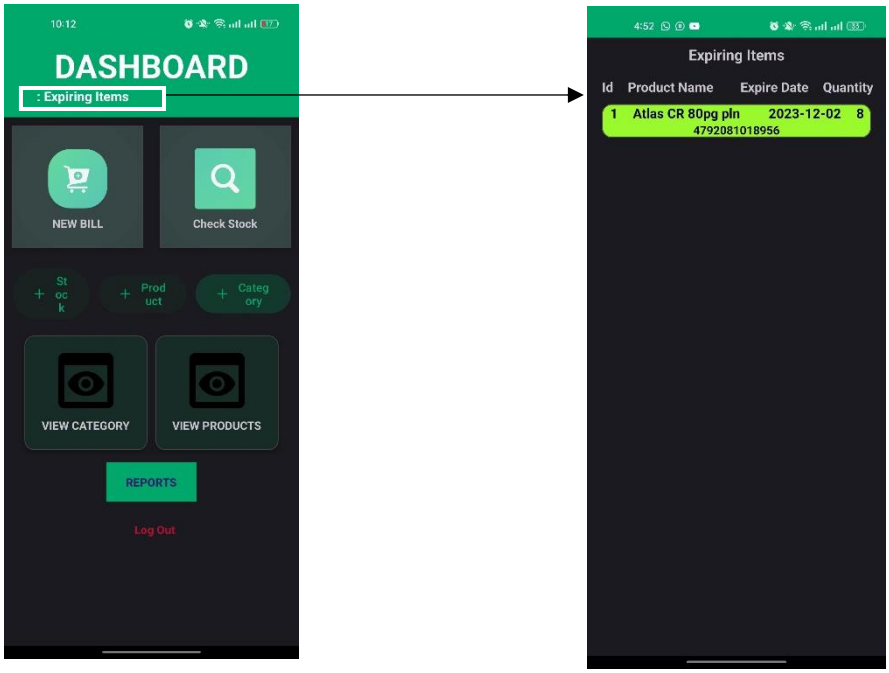
Show Fast selling Items

Show cheques details have to pay



Show averaged buying price & selling prices for each product based on remaining inventories

Expire Items



Challenges

Understand about the Android Anatomy

(1) Android Architecture

App life circle & state changing when app start, pause, save, interrupt, close etc. (State flow & UDF)

Overriding default functions with custom requirements.

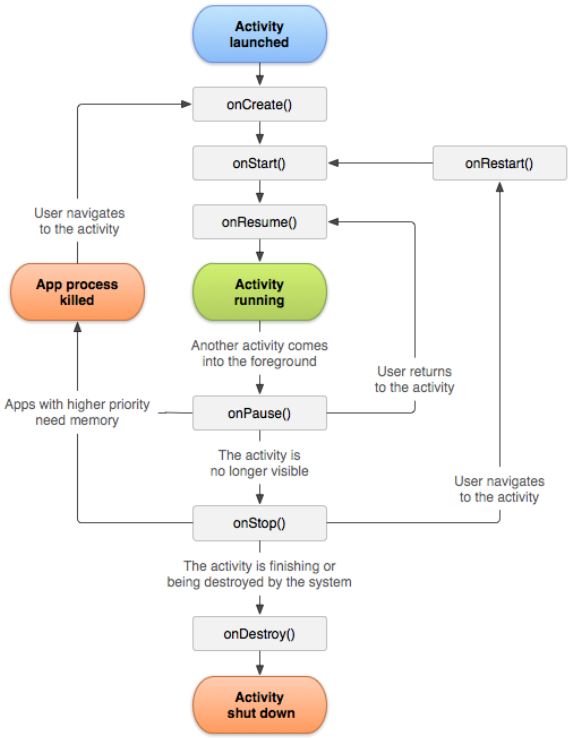


Image 5: android Lifecyle when state changing (Source: developer.android.com)

(2) User Interface Layer:

UI components organize using layouts

Layout Types: Linear Layout

Scroll View

Relative Layouts

Components/ views:

Text related views

Image views

Grouping data (drop boxes/ scroll list)

Button types

User input handle & output viewing

- Customize existing view components, create new templates and work with existing libraries like material designs.

(3) File Structure and Binding UI, Activities, resource files

- Android Manifesto: define, structures, priorities and special access of apps (each UI layout & activity, Default loading page, special accesses need, dependencies etc.)
- Activities classes: Activities related with each UI page & supportive class files.
- Resources files: Images-Icons-Vectors & media files included with APK, Strings, styles & themes
- Builder: Kotlin Engine

(4) Error Handling

Syntax Errors

Logical Errors: specially in conditions and loops.

Runtime Errors: Inputs & output related, Null variables

Semantic Errors: Libraries

Improvements

(1) Add additional functionalities and reports

Report generates for user defined time periods.

Add different analysis features: bucket analysis, trend analysis, foot traffic analysis, Profit margin analysis

Tax calculation

Customers management (Loyalty management)

Communications with Customers (Send bills, promotions)

(2) Improvements for scale up system

Assign different Roles:

Synchronize devices to collaborate working (multi-POS systems)

Cloud synchronization

(3) UI

Improve UI to feel better experience, handle large data sets.

(4) Add revenue models: Options

(I) Subscription with limited free features

(II) Target Advertising in app for selected products to introducing to traders

(III) Pay per use for special features

Project Setup

Copy APK file to device

Run APK file and Install in mobile device