**Software Requirements Specification Document**

Shopping Buddy

Software Requirements Specification

Version 1.1

 11/10/2019

Vennela Cheekoti

Sathwika Gone

Susritha Gade

Rakesh Avirineni

Yaswanth Yarram

Sandeep Devineni

Mehar Choudhary Meenavilli

**Submitted in partial fulfillment**

**Of the requirements of**

**CSIS 44-691 Graduate Directed Project 1**

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
| 06/11/2019 | Version 1.0 | Vennela Cheekoti  Sathwika Gone  Susritha Gade  Rakesh Avirineni  Yaswanth Yarram  Sandeep Devineni  Mehar Choudhary Meenavilli | First Revision |
| 11/10/2019 | Version 1.1 | Susritha Gade  Yaswanth Yarram | Second version |
|  |  |  |  |
|  |  |  |  |

**Document Approval**

The following Software Requirements Specification has been accepted and approved by the following:

|  |  |  |  |
| --- | --- | --- | --- |
| **Signature** | **Printed Name** | **Title** | **Date** |
|  | Dr. Zhengrui Qin | Project Requirements | 06/12/2019 |
|  | Dr. Zhengrui Qin | ER Diagram | 06/21/2019 |
|  | Dr. Zhengrui Qin | Project Schedule(Gantt Chart) |  |
|  | Dr. Zhengrui Qin | UI designs |  |

**Table of Contents                                                                                      Page Number**

 1. Introduction

1.1. Purpose

1.2. Scope

1.3. Definitions, Acronyms, and Abbreviations

1.4. References

1.5. Overview

2. General Description

2.1. Product Perspective

2.2. Product Functions

2.3. User Characteristics

2.4. General Constraints

2.5. Assumptions and Dependencies

3. Specific Requirements

3.1. External Interface Requirements

3.1.1. User Interfaces

3.1.2. Hardware Interfaces

3.1.3. Software Interfaces

3.1.4. Communications Interface

3.2. Functional Requirements

3.3. Use Cases

3.4. Non-Functional Requirements

3.4.1. Performance

3.4.2. Reliability

3.4.3. Availability

3.4.4. Security

3.4.5. Portability

4. Design

4.1. ER diagram

4.2. GUI

5. Analysis Models

5.1. Data Flow Diagram

5.2. Sequence Diagram

**1. Introduction**

**1.1. Purpose**

The purpose of this project is to create a user-friendly e-commerce application which allows the user to shop online and place the order and receive those orders.

**1.2. Scope**

This is an e-commerce application with a login page, if the user is new to the application then he will be allowed to register. User can browse through the different categories (Clothing, Household, Furniture, etc.) and place their order after that user will be redirected to the payment, once payment is done user order is placed successfully and the user will know about when their order will reach. This app also has admin functionality which is to add products to the application so that user can add go through and buy the products, edit /delete products, add/edit/delete promo codes.

**1.3. Definitions, Acronyms, and Abbreviations**

**1.4. References**

**1.5. Overview**

This document is divided into five chapters, Introduction, General Description, Specific requirements, Design and Analysis Models.

         In the Introduction it describe the purpose and scope of the project.

         The General Description gives a clear and concise description of product prospective. It consists of a short view on the product, a short summary of the general capabilities of the product and more in-depth view of the main parts of the product.

         The Specific requirements chapter will describe all the external interface requirements, functional requirements, use cases, class/object definitions and non-functional requirements.  Moreover, here it will describe the user constraints, logical database requirements, and prototype and use case diagrams.

The Design chapter will describe the ER diagram and the UI designs.

The last chapter will describe the Data Flow diagram and Sequence design.

**2. General Description**

**2.1 Product Perspective**

Shopping Buddy is an application used for online shopping developed using open source technologies. This application provides a simple mechanism for users to buy new products and order them online. Some of the main features included in this application are

* User Account: Our application allows the user to create their accounts in the system and provide features of updating viewing their profiles.
* Features & Specifications: The user can get to know about every detail of a particular product.
* Number of users being supported by the application: Though the number is not mentioned but our application supports a good number of users at a time.
* Review: The user is able to review the particular product before the user orders.

Since this is a data-centric product it will need a database to store the data. We add and modify the data whenever needed. All the database communication will go over the internet.

**2.2 Product Functions**

The product functions are divided into two tasks: Admin and User

**Admin Tasks:**

1.  Login

2.  Add product

3. Edit Product

4. Add promo

5. Edit promo

6.  Check Added products, deleted products, added promo, and deleted promo

7.  Inventory (All products)

8.  Chat with user.

**User/Customer tasks:**

1.   Login

2.   Look for products from different categories

3.   Send/reply messages

4.   Change his/her password

5.   Use deals (saves their money)

6.   Share Products

7. View Orders

8. Make Payment

9. Enter Shipping Address

**2.3 User Characteristics**

**·**  Two types of user characteristics: Admin and User/Customer

·   Admin has access to add, edit and delete products whereas user can only view the products and buy products.

·   Admin can add, edit and delete promo codes whereas user can only view those promo codes and use them while purchasing the product.

· User can change his/her password and admin can also change password.

· Admin and user can directly message using messaging characteristics.

**2.4 General Constraints**

The internet connection is a constraint for this application. Since the application fetches the data from the database over the internet, it is important to make sure that there is an internet connection for the application to function.

**2.5 Assumptions and Dependencies**

We assume that the application will always work on mobile phones that have enough performance. If the mobile does not have enough hardware resources available for the application, like if the users might have allocated them with other applications, there may be situations where the application does not work.

Dependencies required for this application are Java language, Android Studio, Firebase, GitHub version control system.

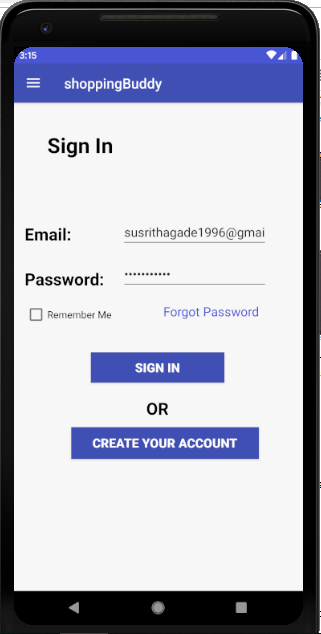
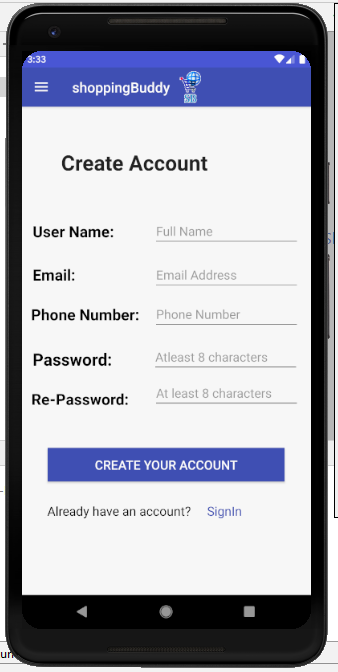
**3. Specific Requirements**

**3.1. External Interface Requirements**

**3.1.1. User Interfaces:**

A good user interface provides a "user-friendly" experience, allowing the user to interact with the software or hardware in a natural and intuitive way. The GUI of our software program includes many controls like the menu bar, Input selector, keyboard, icons, images, and buttons.

User Login Screen                                                      User Registration Screen

**3.1.2. Hardware Interfaces:**

This application works on Android device (Samsung, pixel, etc...). No other hardware is required.

**3.1.3. Software Interfaces:**

Android apps can be written using Kotlin, Java, and C++ languages. The Android SDK tools compile your code along with any data and resource files into an APK, an *Android package*, which is an archive file with an .apk suffix.

* The Android operating system is a multi-user Linux system in which each app is a different user.
* By default, the system assigns each app a unique Linux user ID (the ID is used only by the system and is unknown to the app). The system sets permissions for all the files in an app so that only the user ID assigned to that app can access them.
* Each process has its own virtual machine (VM), so an app's code runs in isolation from other apps.
* By default, every app runs in its own Linux process. The Android system starts the process when any of the app's components need to be executed, and then shuts down the process when it's no longer needed or when the system must recover memory for other apps.

**3.1.4. Communications Interface:**

Google Docs, sheets and slides:

Google Docs, sheets and slides lends itself to collaborative projects in which multiple authors work together in real time from geographically diverse locations. All participants can see who made specific document changes and when those alterations were done. Because documents are stored online and can also be stored on users' computers, there is no risk of total data loss as a result of a localized catastrophe. However, the Internet-based nature of Google Docs, sheets and slides has given rise to concerns among some authors that their work may not be private or secure.

We use this for creating project documents, sheets and slides for presentation. It is reliable for team-work.

Outlook:

It is Microsoft email service. We use outlook to send and receive mails. Used to contact team members.

**3.2 Functional Requirements**

**Admin Requirements**

**1.   Admin login**

Administrator should be able to log in to the system with admin privileges using his admin username and password

|  |  |
| --- | --- |
| Admin username | shoppingbuddyseven@gmail.com |
| Password | Minimum of 6 characters  At least one number or symbol (!, #, %, &. \*)  (admin7) |

**2. Add new products:**

Administrator should be able to add new items to the list. Most recently added item should be displayed on the top of the inventory list. New item should have following fields.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Product ID | Product Name | Description | Category | Cost | Image |

**3. Edit/Remove Products:**

Administrator should be able to edit or remove any existing items on the list. When editing, he can edit all the fields except ID.

**4. Add new promo Codes:**

Administrator should be able to add new promo codes to the list. Most recently added item should be displayed on the top of the inventory list. New item should have following fields.

|  |  |  |  |
| --- | --- | --- | --- |
| Promo ID | Product Code | Description | Amount to Deduct |

**5. Edit/Remove Products:**

Administrator should be able to edit or remove any existing promo code on the list. When editing, he can edit all the fields except ID.

**6. Check Added products, deleted products, added promo, and deleted promo:**

Administrator should be able to view the products that he/she added and deleted and also promo codes that are added and deleted

**7. Inventory (All products):**

Administrator should be able to see all the products that added are to the application in purpose for the users to buy.

**8. Chat with User:**

Administrator should be able to chat with user if in case user messages.

**User Requirements:**

**1.   Register:**

When a potential user wants to buy anything from Shopping Buddy, he/she should fill the sign up form. When a user registers for the system, he/she should provide the following details.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User Name | Email | Phone Number | Password | Re- Enter Password |

**2. Log in to Shopping Buddy - Android**

Once user is done with registration then user can log in to the Shopping Buddy– Android using his email address and password.

**3. Recover lost or forgotten password**

If user lost or forgot the password, he can recover the password by entering his/her email then the link to reset password will be sent to their email.

**4. View the Products and details**

Once the user log in to the Shopping Buddy– Android, he/she can view the list of products of different categories. In the initial list it shows the product name and price.

|  |  |
| --- | --- |
| Product Name | Product Price |

 If user wants to see more details about a specific product, he can tap on the product and view other information about the products.

**5. Share Products:**

User can share the products he/she likes with others, so that even others can give their opinion about the product and even buy if they need the same product.

**6. Add to cart:**

User can add the products to the cart and can buy the products.

**7. Shipping Address and Payment:**

User can enter his/her shipping address and then he/she can do payment in order to buy the product. If the payment is done then the user is done with purchasing the product.

**8. Orders:**

User can view all the orders that he/she has done.

**9. Settings:**

User can change his/her password, and can also contact admin by using the email or he/she can message admin.

**10. Logout**

**3.3. Use Cases**

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. A use case can be thought of as a collection of possible scenarios related to a particular goal, indeed, the use case and goal are sometimes considered to be synonymous.

A use case (or set of use cases) has these characteristics:

·     Organizes functional requirements

·     Models the goals of system/actor (user) interactions

·     Records paths (called *scenarios*) from trigger events to goals

·     Describes one main flow of events (also called a basic course of action), and possibly other ones, called *exceptional* flows of events (also called alternate courses of action)

·  Is multi-level, so that one use case can use the functionality of another one.

**3.4. Non-Functional Requirements**

**3.4.1. Performance:**

Requirements about resources required, response time, transaction rates, throughput, benchmark specifications or anything else having to do with performance.

**3.4.2. Reliability:**

Requirements about how often the software fails. The measurement is often expressed in MTBF (mean time between failures). The definition of a failure must be clear. Specify the consequences of software failure, how to protect from failure, a strategy for error detection, and a strategy for correction.

**3.4.3. Availability:**

Availability is gauged by the period of time that the system’s functionality and services are available for use with all operations.

**3.4.4. Security:**

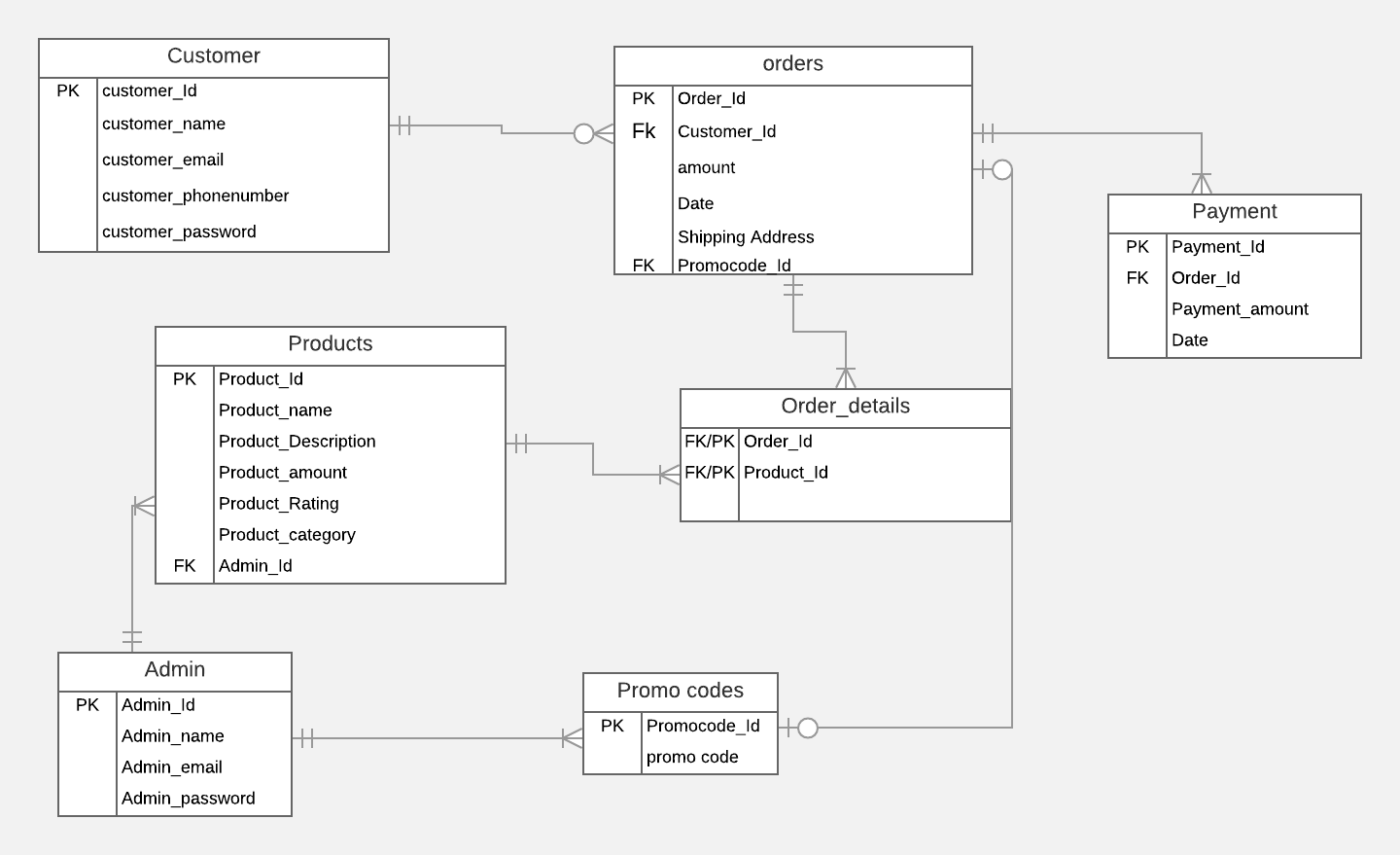
One or more requirements about the protection of your system and its data. The measurement can be expressed in a variety of ways (effort, skill level, and time) to break into the system.

**3.4.5. Portability:**

The effort required to move the software to a different target platform. The measurement is most commonly person-months or % of modules that need changing.

**4. Design**

**4.1 ER Diagram:**

****

In the above Entity Relationship diagram, it has 7 entities namely Customer, Products, Admin, Orders, Order\_details, Promo codes and payment. The customer can be considered as user and admin can be considered as seller. From the above diagram we can also see that we have only one associative entity that is Order\_details which is a weak entity. Customer\_id, Product\_id, Admin\_id, Order\_id, Promocode\_id, Payment\_id are primary and foreign keys.

**Relationship:**

         Customer: Orders               - 1: M

         Orders: Payment                 - 1: M

         Orders: Order\_details         - 1: M

         Product: Order\_details        - 1: M

         Orders: Promo codes          - 1:1

         Product: Admin                  - 1: M

         Admin: Promo codes          - 1:M

4.2 GUI

|  |  |
| --- | --- |
| Login Page | Forgot Password |
| Capture |  |
| Register | Home Screen |
| Captureq |  |
| Item Details | Cart |
|  |  |
| Shipping Address | Payment |
|  |  |
| Orders | Settings(Change Password) |
|  |  |
| Admin Login | Admin Home |
|  |  |
| Add Product | Edit Product |
|  |  |
| Add Promo Code | Edit/Remove Promo Code |
|  |  |
| Admin Inventory | Admin |
|  |  |

**5.** **Analysis Models**

## 5.1 Gantt Chart

Please refer to the support document Gantt Chart PDF and Gantt Chart MPP.

Shopping Buddy - Android App

Version 2.0.0

# Technical Manual

Team : Android Team

Course: 44692-04

Class : GDP 02 Spring 2019

TEAM MEMBERS  
Sathwika Gone  
Vennela Cheekoti  
Rakesh Avirineni

Yaswanth Yarram

Mehar Choudhry Meenavilli

Sandeep Devineni

Susritha Gade

# 1 Introduction

## 1.1. Document Identification

This document describe the technical aspect of design and implementation of Shopping Buddy App for android based mobile phone. This document is prepared by Android team of GDP 01 (44692-04) Fall 2019.

## 1.2. System overview

Shopping Buddy -Android is developed for android base mobile devices to make it possible for user to buy the products from different categories which are added to application by admin. This app consists of two parts those are admin and user. Users can download the application through internet and use it to by the products. Admin will use the applications to add the products, reply to user messages/chats and other administrative tasks.

## 1.3. Tools and technologies

### 1.3.1. Android Studio 3.2.1

Android Studio 3.2.1 is used to develop the User Interfaces (UIs) -Activities, and frontend and backend functionalities.

### 1.3.2 Firebase

Google Firebase is used for backend data handling, storing and manipulation. Also Authentication, Email verification and password recovery functionalities are implemented using firebase functionalities.

Firebase Account Credentials

User Name :shoppingbuddyseven@gmail.com

Password : sevenbuddies

### 1.3.3 GitHub

Github repository is used for version control and collaborating purpose of the development process. All the implementation and documentation work were uploaded to the github by each member of the team as they work.

Github link : <https://github.com/SathwikaGone/Shopping-Buddy>

### 1.4 Document overview

This document will describe the technical aspect of each activity, functionality and associations separately. The list and order of the items will be as follows.

1. Sign Up for user
2. Login for user
3. Password recovery for user
4. User logout

Documentation will be continued as the functionalities are implemented….

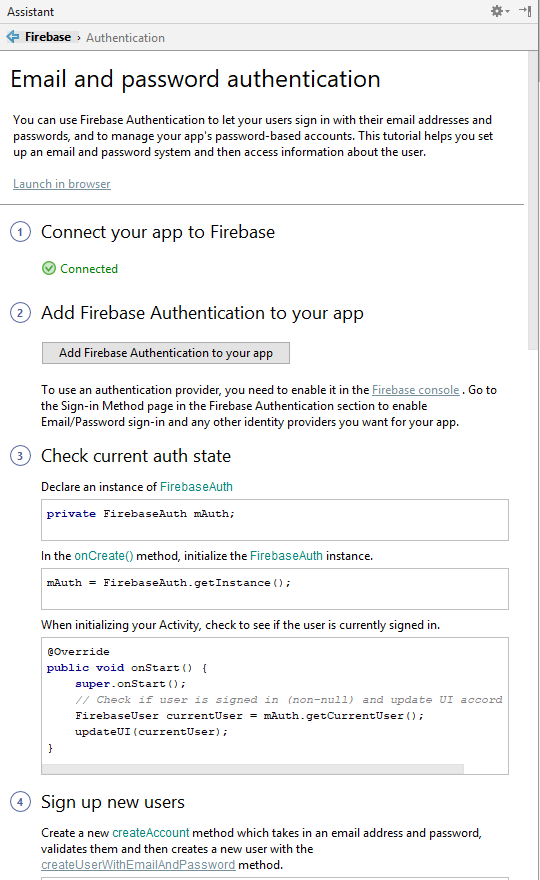
# 2. User Signup

## 2.1. Requirement

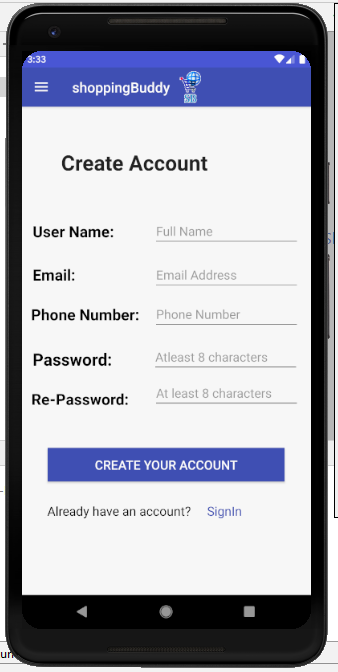
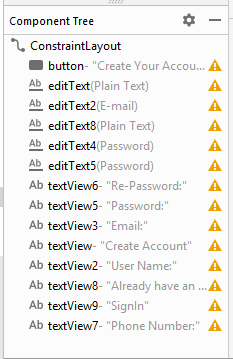
Once the user download the app, he/she can see the registration page. Then user can provide the necessary information and sign up for the app. Then the user will get a verification email to his email account and he must verify his email in order to get access to the app.

## 2.2 Technical information

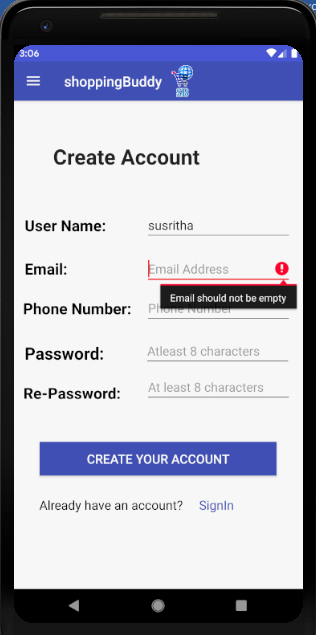
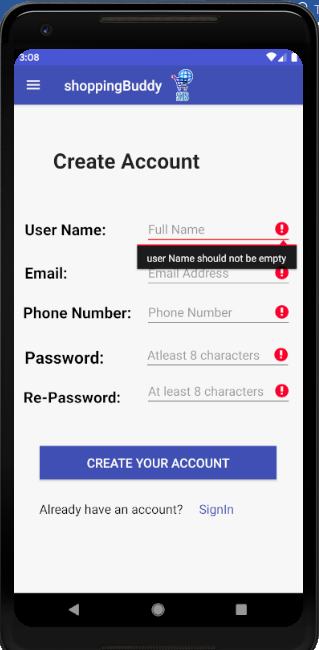
i. App is connected to the Firebase and activated the Authentication with email and password authentication.



ii. Signup activity is created on Android studio

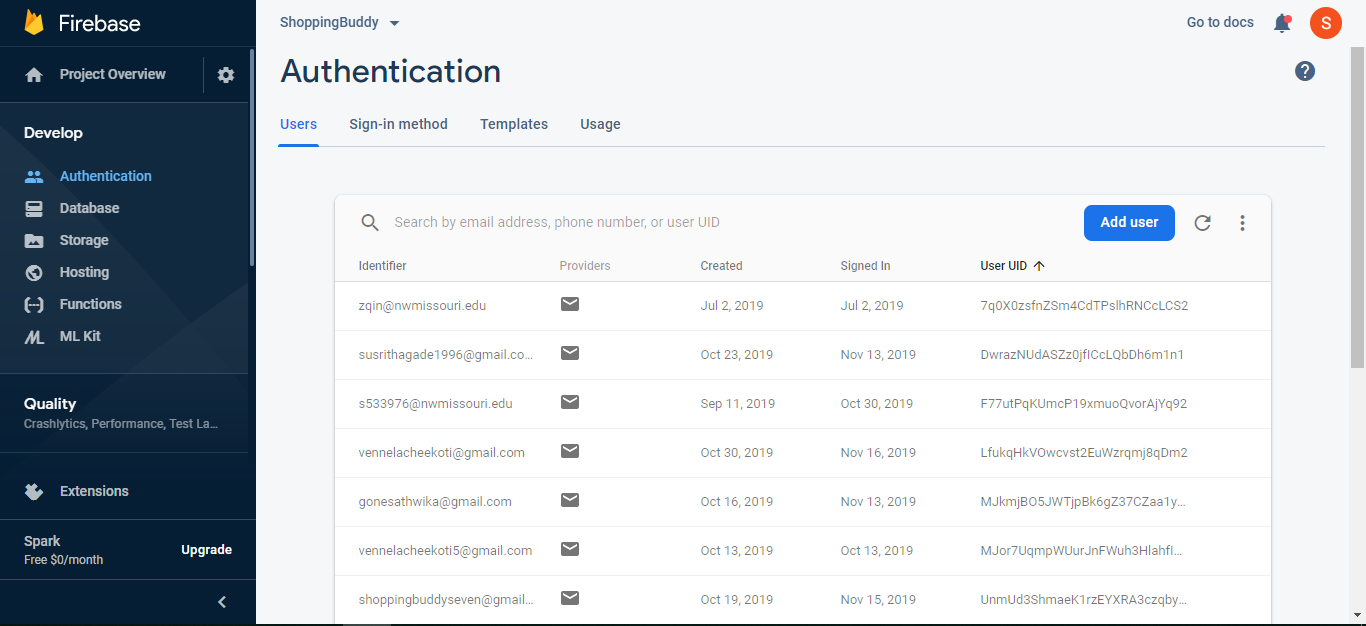
iii. User need to fill all the information on signup page, otherwise it will send message stating that field should not be empty.

iv. Once user fill all the information and click on continue, it will create a record on the firebase authentication and will send an email to the user for verification.

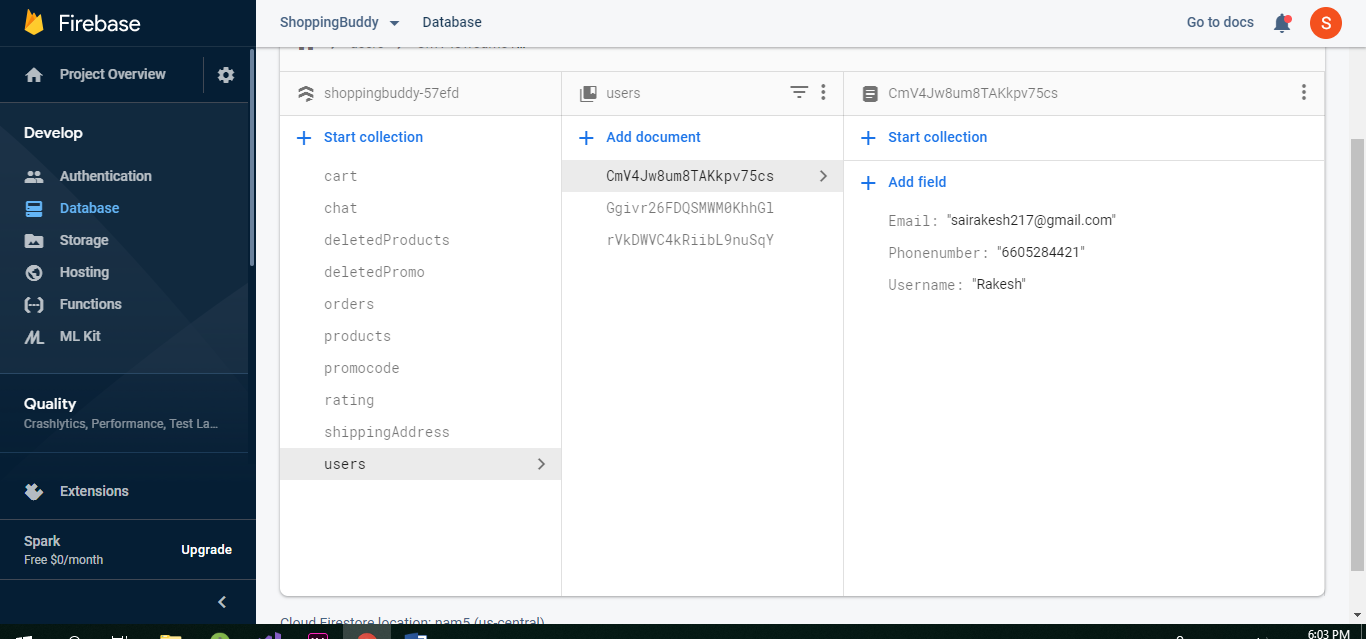


vi. Authentication data will store in the firebase authentication and personal data will store on firebase firestore.



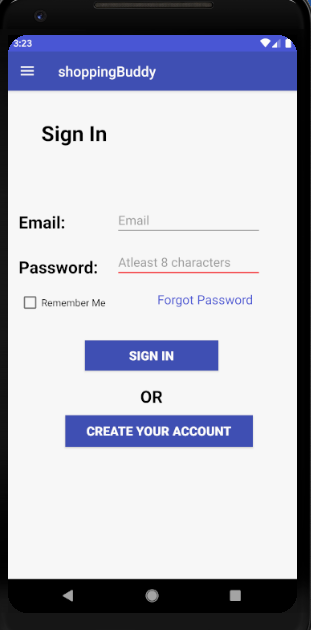
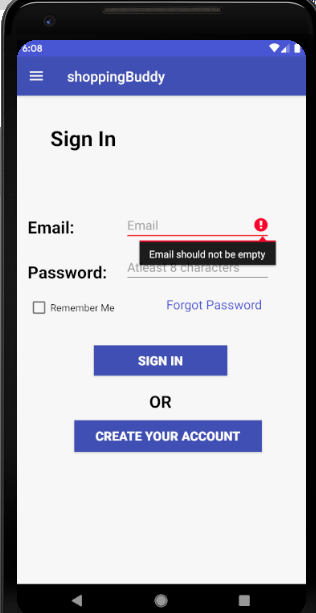
In firestore, user details will store as follows.

Here document id would be the user’s email address.



3. User Login

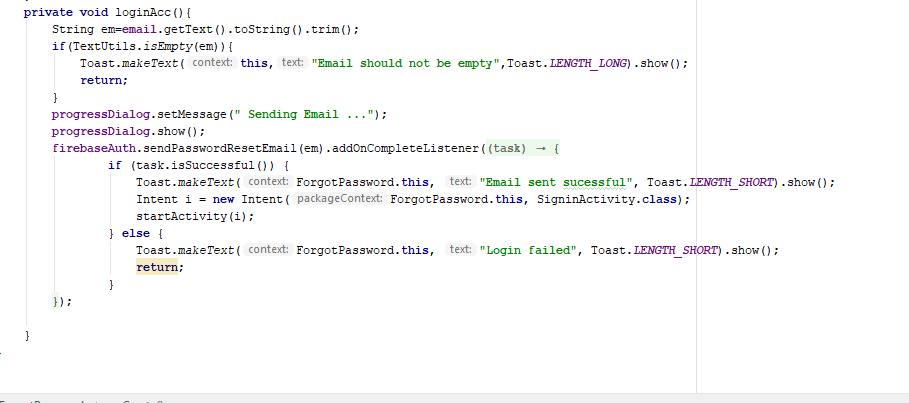
If the user haven’t entered username or password, it will display the message in toast.

 Login Functionality:

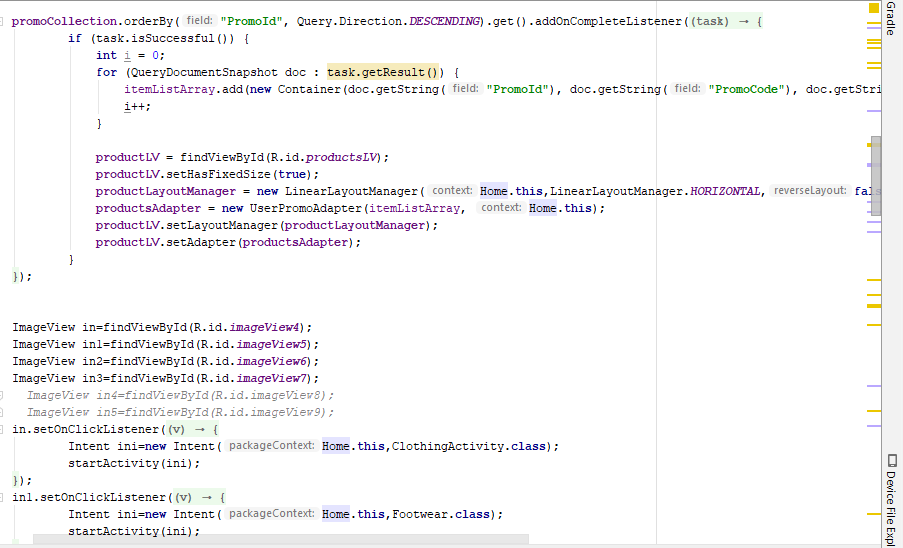


Password Recovery for User:

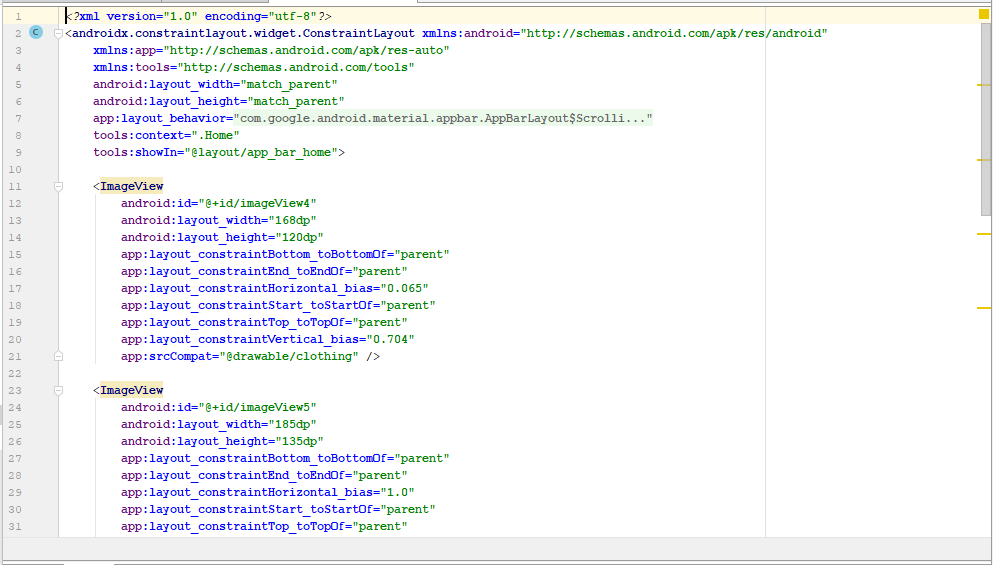
If the user forgets his/her password, he/she can request for the change of password using forgot password option.

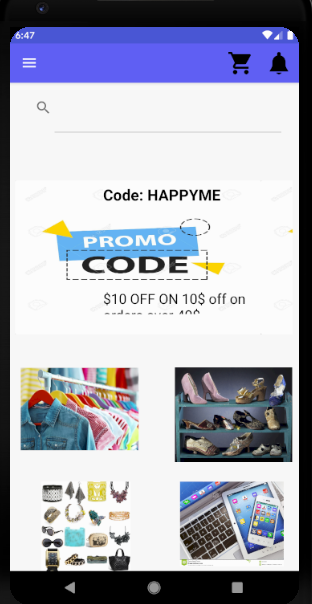


Home page:



Xml Code:





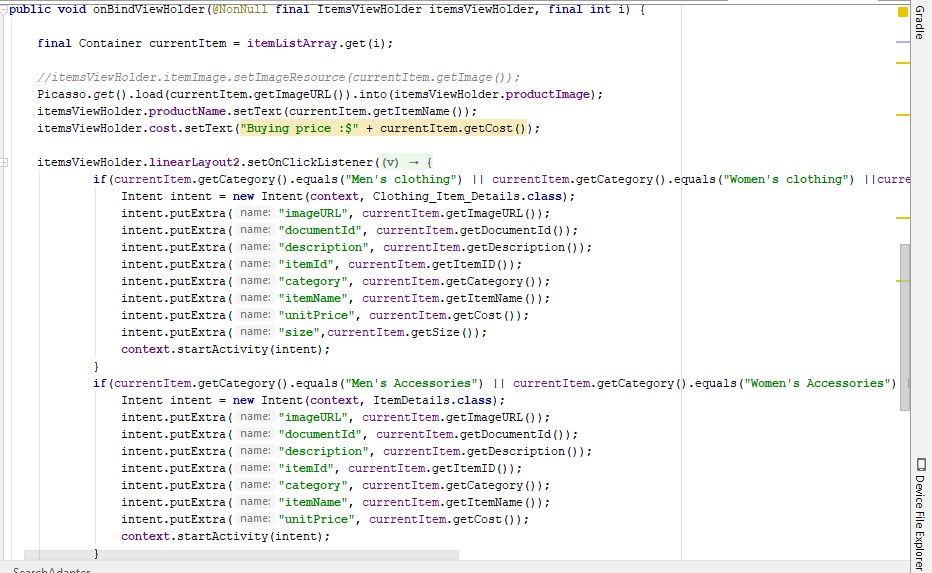
Search Functionality:

User can search various products through this search functionality, if he/she doesn’t want to go through the categories and want to find the product quickly.

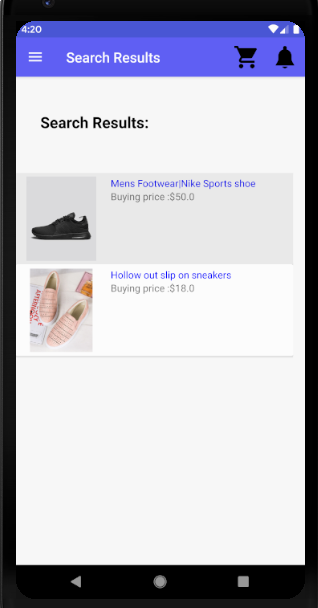


Adapter:





Phone View



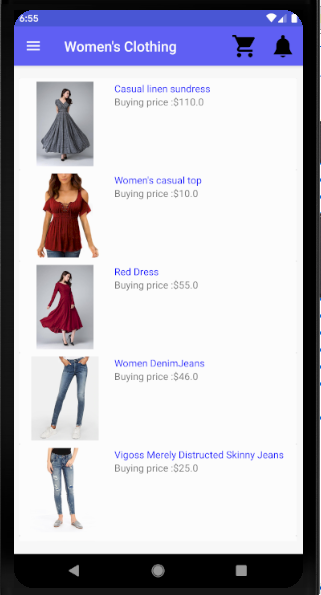
Item list Functionality:



Xml Code:



Phone View:

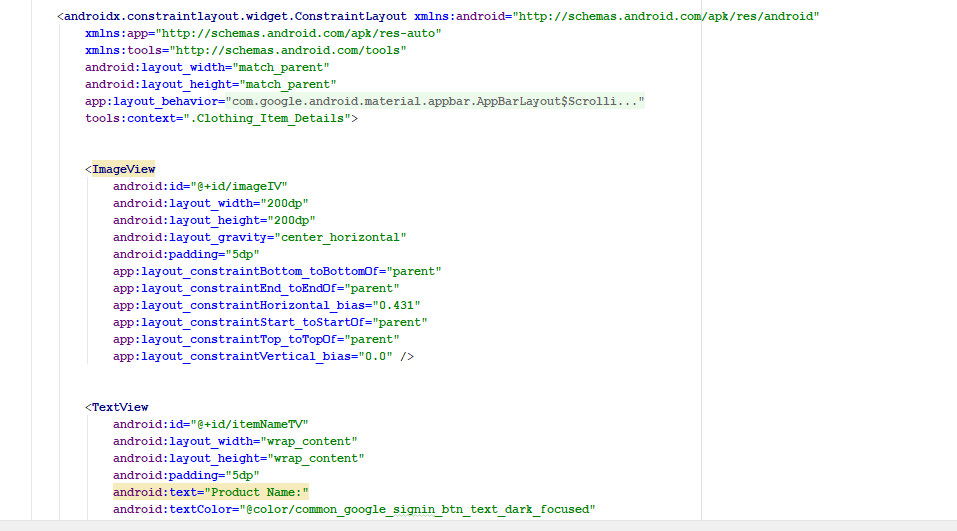


Products Detail Page:





Xml Code:





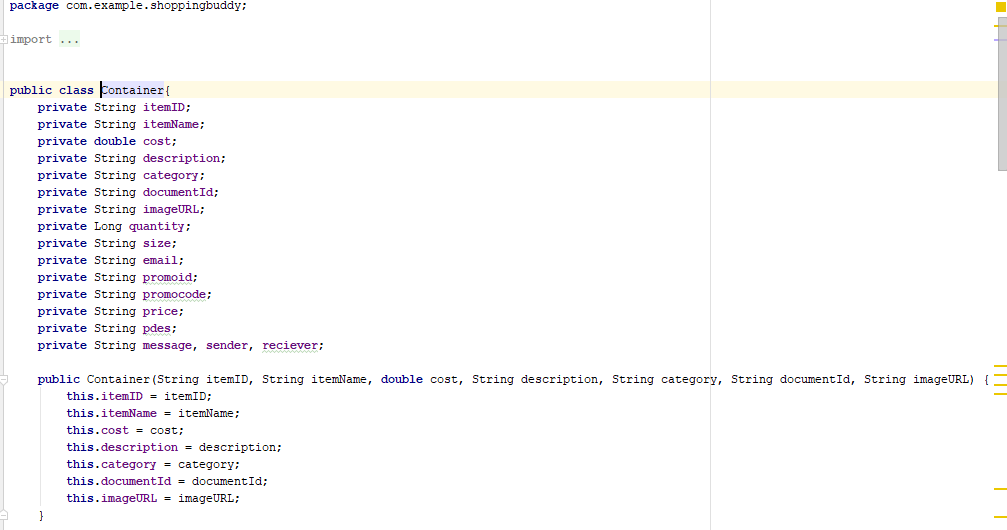


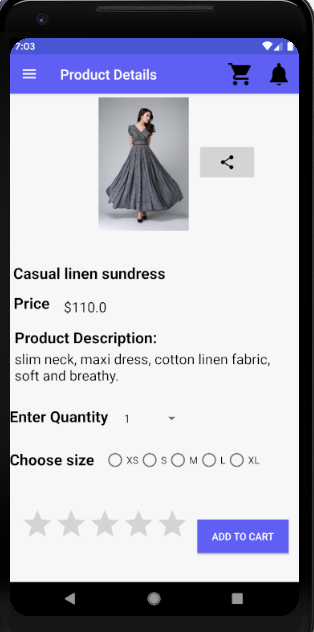












Cart Functionality:

Whenever user adds the products to the cart they are stored and he/she can buy the products by clicking on checkout and can also delete the products from cart if he/she doesn’t like the product anymore. All the products are added to the firebase database



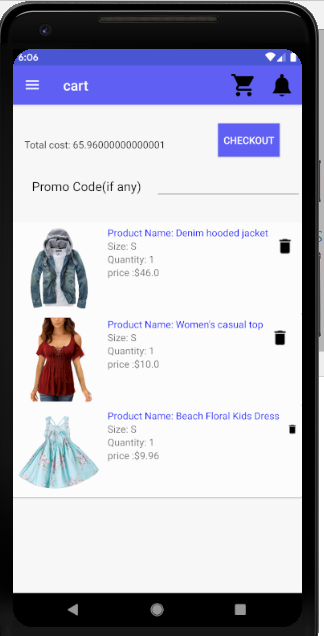


Xml code:









Shipping Address:

User need to enter the address to which he wants his order to be delivered and shipping address is added to the firebase database.

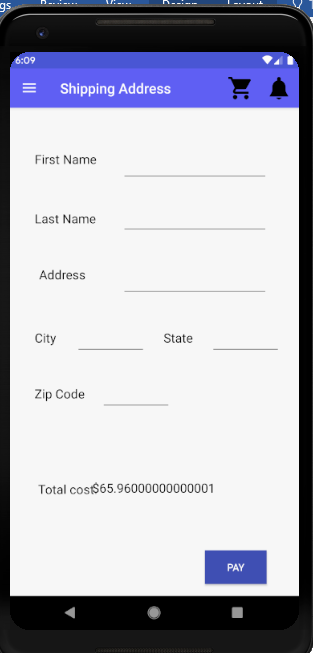


Xml code



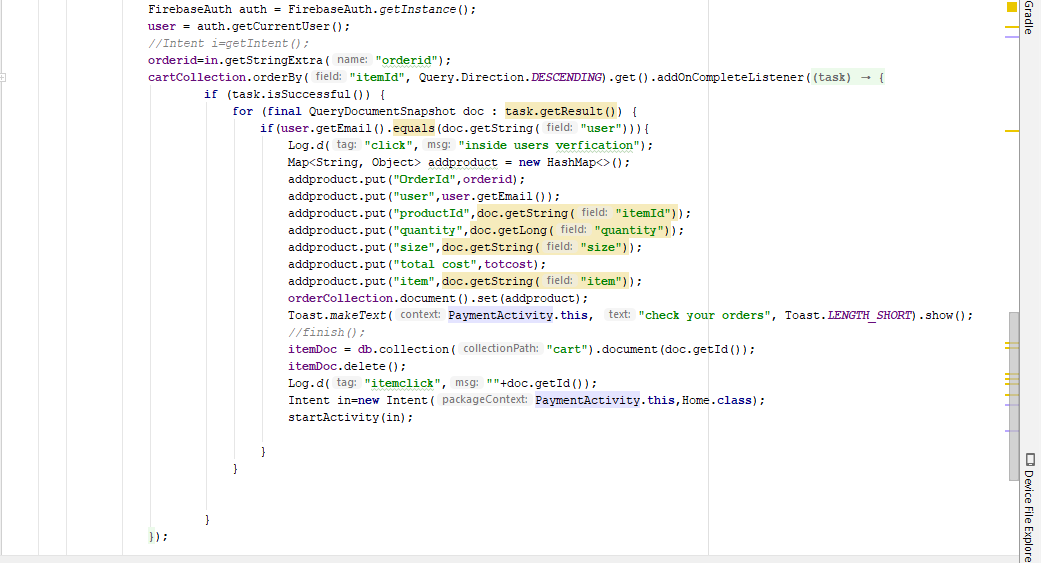


Phone view:



Payment:

After entering the shipping details then user can do his/her payment. Payment details will be stored in the firebase.



Xml code

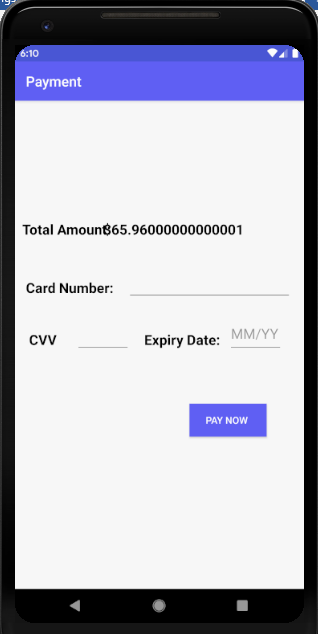








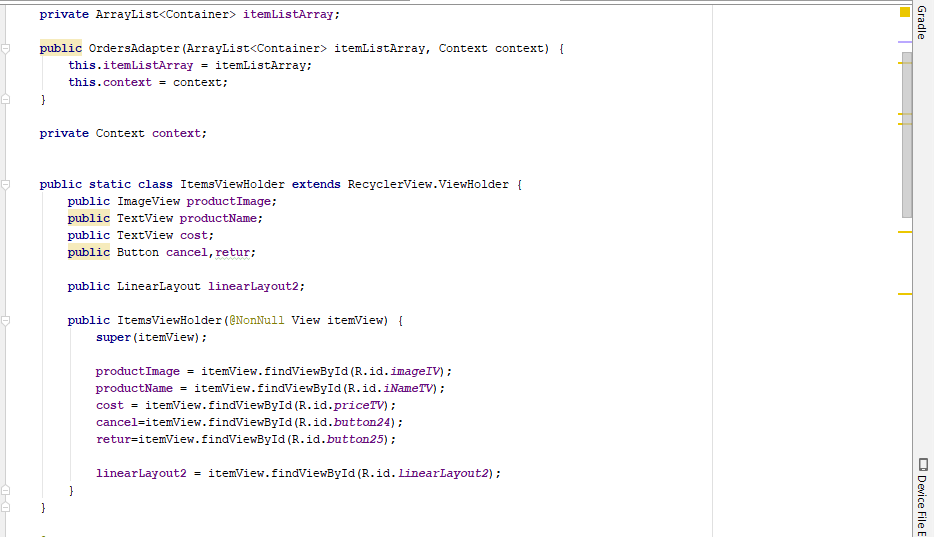
Phone view:



Orders:

After the payment is done then user can look at the orders he/she had made and can cancel or return products.

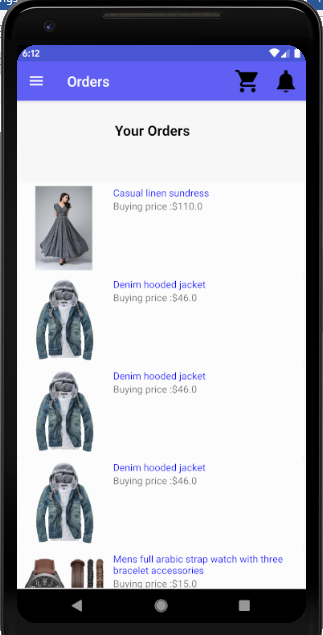




Xml code:



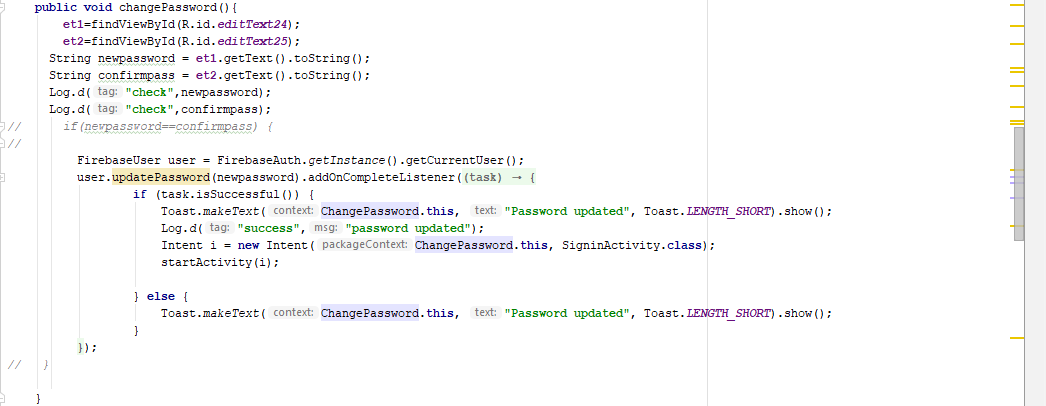
Phone view



User Settings:

User can change his privacy settings like changing password and can also contact us.

Privacy:



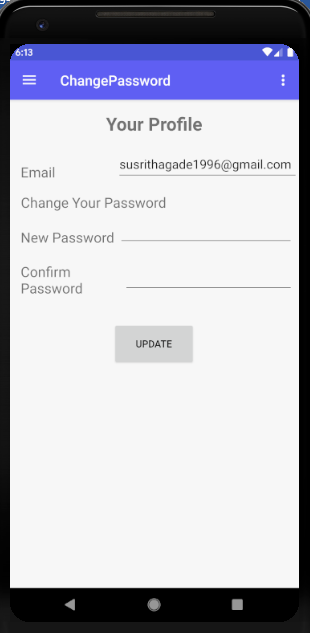
Xml code:





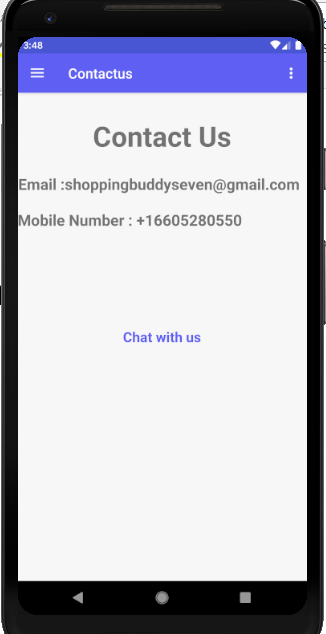


Phone view

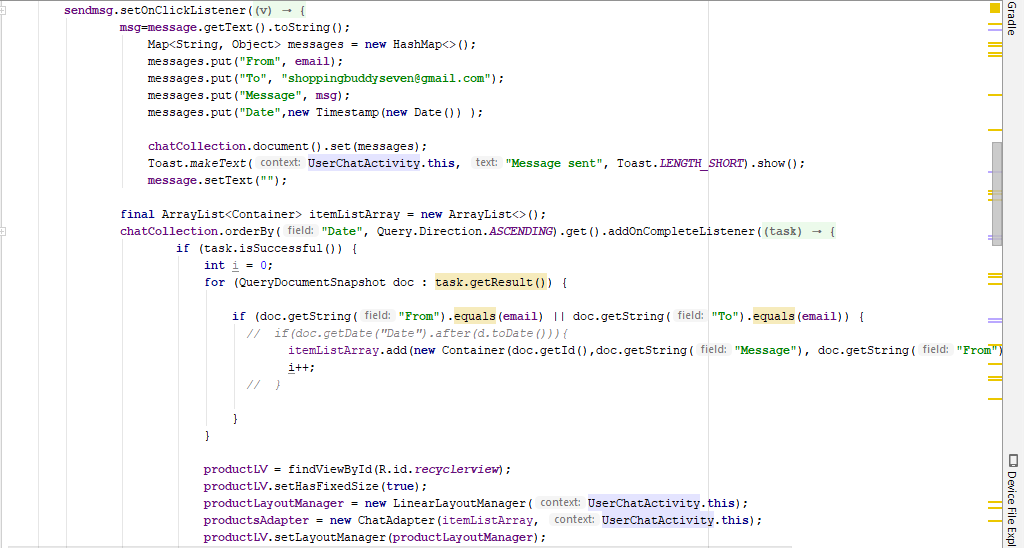


Contact us:

User can contact admin either by mailing or he/she can directly chat with admin.



Chat:

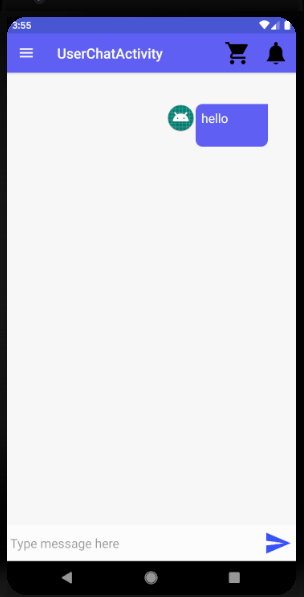




Xml code



Phone View:



Shopping Buddy- Android App

Version 2.0.0

# **Technical Manual-Admin**

Team : Android Team

Course: 44692-04

Class : GDP 02 Spring 2019

TEAM MEMBERS  
Sathwika Gone  
Vennela Cheekoti  
Rakesh Avirineni

Yaswanth Yarram

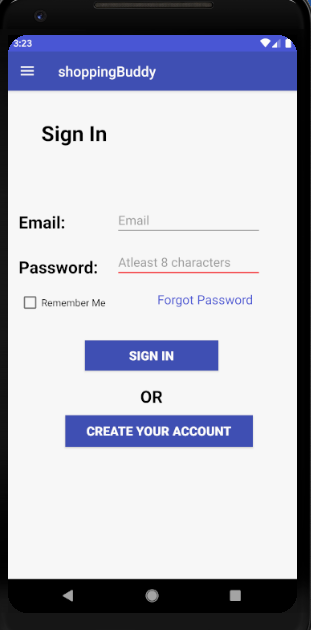
Mehar Choudhry Meenavilli

Sandeep Devineni

Susritha Gade

Admin Sign In functionality:

Admin can sign in into the application with the email that he/she has been assigned and by entering password.





Admin Home:

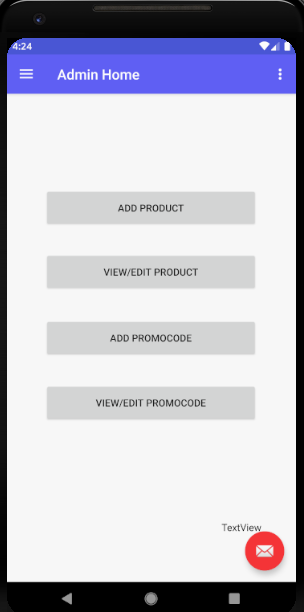


Xml code





Phone view



Add Product:

Admin has to add the products so that the user can buy products and all these products are stored in the firebase database.



Xml Code:

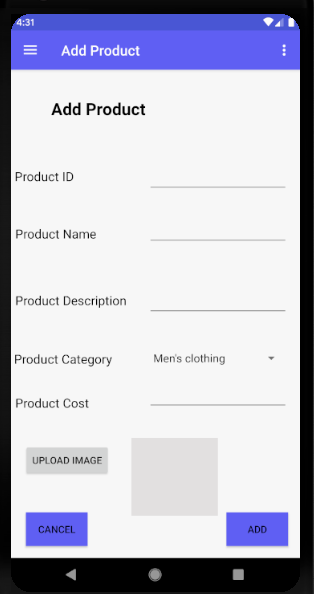




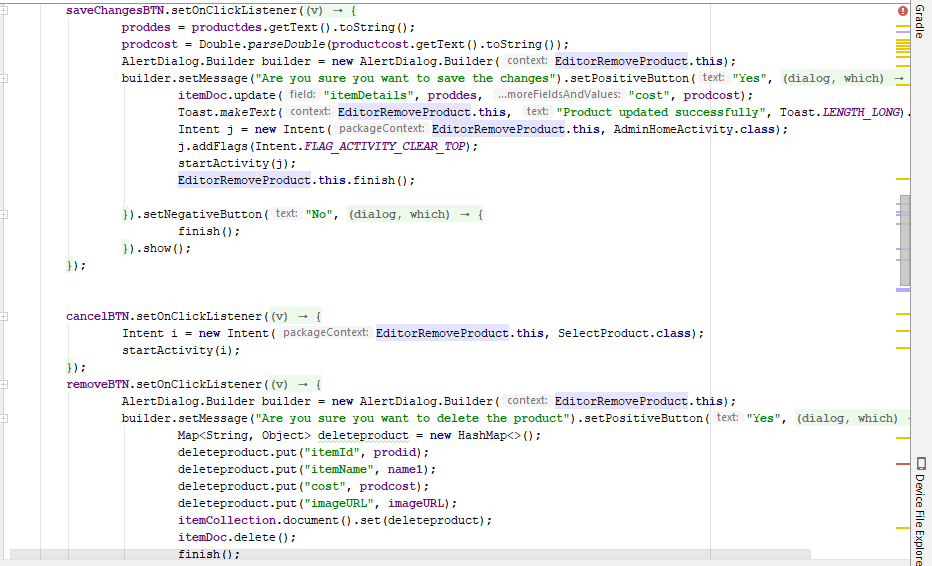




Phone View



View/Edit Product: Admin can Search for the product that he/she wants to edit or view the product after the products get displayed admin can edit the product. Admin can also remove product.



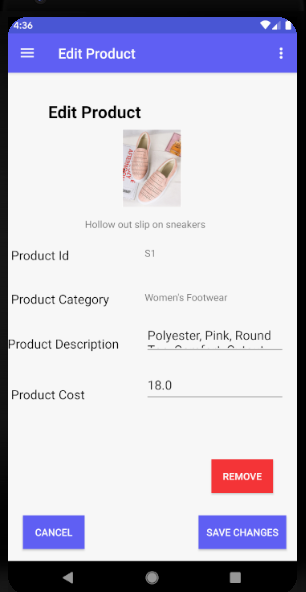
Xml code





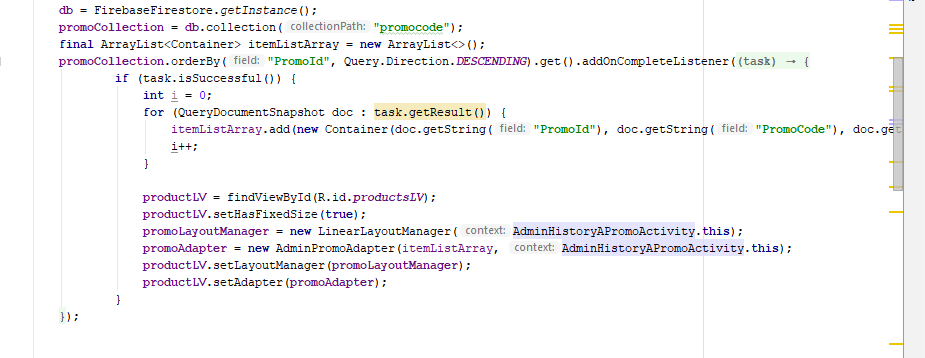


Phone View



Add promo Code

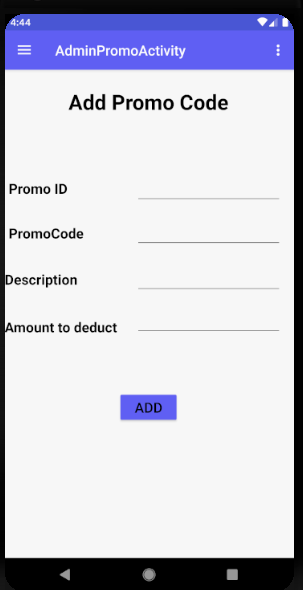
Admin can add the promo code so that users can use the promo code and get some discount when the deal is active.



Xml Code



Phone View



View/Edit Promo Code

Admin can edit the promo code and can also remove the promo code.

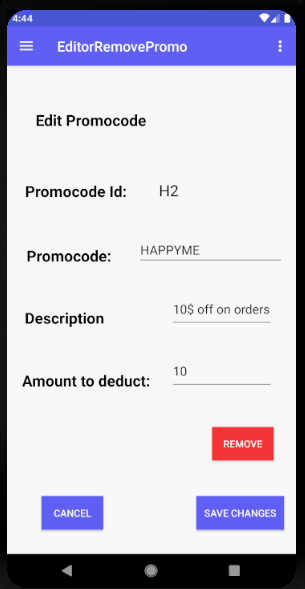


Xml Code





Phone View

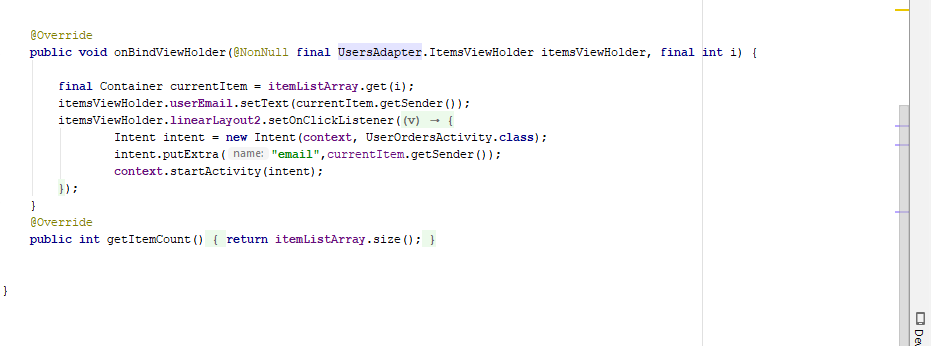


User History:

Admin can view the list of the users for the application.

Adapter:





Phone View

