**PROJECT REPORT**

**on**

**E-Commerce Mobile App Development**

**(CS V I Semester mini project)**

**2018-19**

****

**Submitted to: Submitted by:**

Mr. B. P. Dubey Mr. Rahul Singh

**Guided by:**  Mr. Harish Singh Chauhan

Mr. Avnish Panwar Mr. Pankaj Rawat

**DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY**

**GRAPHIC ERA HILL UNVERSITY, DEHRADUN**

**CERTIFICATE**

### Certified that Mr. Rahul Singh (Roll No.- 1018140), Mr. Pankaj Rawat (Roll No.- 1018122) and Mr. Harish Singh Chauhan (Roll No.- 1018071) have developed mini project on “Ecommerce Mobile App development” for the CS VI Semester mini project Lab (PCS604) in Graphic Era Hill University, Dehradun under my supervision. The project carried out by Students is their own work as best of my knowledge.

Date:

Mr. Avnish Panwar

Project Guide

Assistant Professor

Department of CS/IT GEHU, Dehradun

**ACKNOWLEDGMENT**

We would like to express our gratitude to The Almighty Shiva Baba, the most Beneficent and the most Merciful, for completion of project.

We wish to thank our parents for their continuing support and encouragement. We also wish to thank them for providing us with the opportunity to reach this far in our studies.

We would like to thank particularly our project Co-ordinater Mr. B. P. Dubey and our Project Guide Mr. Avnish Panwar for his patience, support and encouragement throughout the completion of this project and having faith in us.

We also acknowledge to all teachers specially Shashi Sharma Sir, Risika ma'am, Avinish Sir and Amit Gupta Sir who gave valuable suggestions to us in developing the project.

At last but not the least We greatly indebted to all other persons who directly or indirectly helped us during this work.

**Mr. Rahul Singh Mr. Harish Singh Chauhan**

**(Roll No:1018140) (Roll No: 1018071)**

**Mr. Pankaj Rawat**

**(Roll No: 1018122)**

TABLE OF CONTENTS

**CHAPTER NO. TITLE PAGE NO.**

**LIST OF FIGURES vi**

**LIST OF SYMBOLS, ABBREVIATIONS vii**

**1. INTRODUCTION** 8

1.1 About Project 8

1.1.1Abstract 8

1.1.2 Objectives of organization 8

1.1.3Analysis of present system 8

1.1.4Problem of existing system 9

1.1.5Features of proposed system 9

1.2 Android Phone Application Platform 10

1.3 Android Development Framework 11

1.3 Android Software Stack 11

**2. PROJECT 12**

2.1 Requirement Analysis 12

2.1.1 Hardware Requirements 12

2.1.2 Software Requirements 12 2.1.3 Secondary Requirement 13

2.2 Algorithm Flow Chart 13

2.3 Use cases 14

2.4 Datbase Structure 16

2.5 Modules 20

2.5.1 Administrator Module 20

2.5.2 Customer Module 20

**3. SNAPSHOT OF PROJECT 22**

3.1 sign up page 22

3.2 Login by Admin 23

3.3. Login by User 24

3.4. OTP Generation 25

3.5. search 27

3.6. Add to cart 27

3.7 Drawer 28

3.8Payment 28

**4. CONCLUSION 30**

4.1 SUMMARY 30

4.2 FUTURE WORKS 30

**APPENDIX:** CODE 31

**REFERENCE** 69

**LIST OF FIGURES**

**FIGURE Page No.**

Figure 1.1 Algorithm Flow Chart 12

Figure 1.2 Use Case Diagram 13

Figure 2.1 Database structures 16-19

Figure 2.2 Snapshot of Project 22-29

**LIST OF SYMBOLS, ABBREVIATIONS**

∪ Union

∩ Intersection

⊆ Subset or Same

∞ Infinite

DBMS Data Base Management System

SQL Structure Query Language

**CHAPTER** **1**

**INTRODUCTION**

* 1. **ABOUT PROJECT**

1.1.1 ABSTRACT

E-Commerce has improved business methods by giving the businesses the opportunity of selling goods and services on a universal basis. The Internet provides traders the possibility of spreading out their shops into infinite number of sites, and also gave the consumers the advantage of shopping across all borders. To give the customers not only the advantage of buying anytime, but also the advantage of buying anywhere, using a mobile device for E-commerce has become an alternative [.

Progression of wireless technology made mobile devices more popular. From the customers’ point of view in one hand, it is advantageous to use mobile devices such as smart phones and take advantage of their mobility while surfing an online store, while in the other hand the shop owners benefit from fulfilling the customer’s desires. This new method of purchasing, which is more convenient for the consumers and more profitable for the shop owners, is called mobile commerce or in abbreviation m-commerce

.

1.1.2 OBJECTIVES OF ORGANIZATION:

E-Commerce has improved business methods by giving the businesses the opportunity of selling goods and services on a universal basis. The Internet provides traders the possibility of spreading out their shops into infinite number of sites, and also gave the consumers the advantage of shopping across all borders. To give the customers not only the advantage of buying anytime, but also the advantage of buying anywhere, using a mobile device for E-commerce has become an alternative.

Progression of wireless technology made mobile devices more popular. From the customers’ point of view in one hand, it is advantageous to use mobile devices such as smart phones and take advantage of their mobility while surfing an online store, while in the other hand the shop owners benefit from fulfilling the customer’s desires. This new method of purchasing, which is more convenient for the consumers and more profitable for the shop owners, is called mobile commerce or in abbreviation m-commerce .

1.1.3 ANALYSIS OF PRESENT SYSTEM

Before we begin a new system, it is important to study the system that will be improved or replaced (if there is one). We need to analyse how this system uses hardware, software, network and the people resources to convert data resources, such as transaction data, into information products, such as reports and displays. Thus, we should document how the information system activities of input, processing, output, storage and control are accomplished.

1.1.4 PROBLEM OF EXISTING SYSTEM:

The problem of browsing online stores with smart phones is that, these web shops are designed to be used with a device of big size screen and their functionality are optimized for those kind of devices and in comparison, smart phones have a much smaller screen size. As a result, the navigation process of listing the categories, selecting a product, viewing the details, adding it to the shopping cart and issuing the order on the mobiles web browser is pretty tough, boring and time consuming.

1.1.5 FEATURES OF PROPOSED SYSTEM:

The solution could be using the mobile device features in a suitable way for web shops to be optimized for that size of screen. Taking the same functionality and adopting them to be used on a smart phone application could lead to a much better experience than using the native web browsing option.

There are a lot of online stores, but instead of writing an application for each of them which is expensive, one can write a generic client to be compatible with a bunch of them ,this allows to reduce cost significantly.

These online stores are usually generic instances of a framework, so the abstract client, which has been written for the abstract web shop, could in the end be specialized and instantiated for different framework instances (actual web shops). The client's instantiating cost would then be, setting the specific web shop's Internet address and branching.

**1.2 Android Phone Application Platform**

The important features about the use of Android as a development environment are centred around the Application Programming Interface (API) it provides. Android is an application-neutral platform, which provides the opportunity to create applications that have access to the device hardware through series of APIs libraries. The following are common Android features:

Ø Free development, distribution and licensing

Ø Wi-Fi hardware access

Ø API libraries for implementing accelerometers, compass, camera, Bluetooth and

location-based services like GPS (Global Positioning System).

Ø Framework for localization

Ø Support for 2D and 3D graphics using OpenGL ES 2.0 (Open Graphics Library

for Embedded System)

Ø SQLite Database for storage and retrieval

Ø Media API that supports playing and recording audio and video format

Ø Open-source HTML5 (HyperText Markup Language) Webkit-based browser

Ø Shared data possibility through content providers, intents and notifications

Ø Support for background services for processes and applications

Ø Provision for application components’ reuse and the replacement of generic applications.

Ø Memory and process management.

* 1. **Android Development Framework**

Each Android application is written in the Java Programming language and run on a separate instance of a virtual machine called Dalvik Virtual Machine (DVM) contrary to the Java Virtual Machine (JVM). The Android runtime is responsible for the memory and process management, which kills the process when the need arises to release memory to the Android runtime. The Android runtime and the Dalvik Virtual Machine sit

on a Linux Kernel that interacts with the low-level hardware. A set of APIs is then necessary to expose the underlying hardware features and services.

**1.4 Android Software Stack**

Android applications run on top of a Linux Kernel and a collection of C/C++ libraries that provide applications with hardware functions of the device: hardware drivers, memory management and power management. The Linux Kernel is also responsible for process management, which starts a separate process for each application and

multiple threads can execute the application in the same process.

**CHAPTER** **2**

**PROJECT**

**2.1 REQUIREMENT ANALYSIS**

2.1.1 Hardware Requirement

RAM: 2 GB.

PROCESSOR: Intel Core I5 7th Generation.

HARD DISC: 500GB.

The device must support for minimum network capability (EDGE - Enhanced

Data Rates for Global Evolution, HSPA – High Speed Packet Access, EV-DO –

Evolution Data Optimized, 802.11g and Wi-Fi)

The device must have at least 128MB of memory available to the Linux Kernel

The device must have at least 1GB of non-volatile storage for user data

The download Manager capable of downloading individual files of at least 55MB

in size

The device must implement at least a soft keyboard for user input

The device must have a touchscreen (capacitive or resistive touchscreen)

Support for OpenGL ES 1.0

The device must support dynamic orientation by application to either portrait or

landscape screen orientation

2.1.2 Software Requirement

1. Languages Used: Android,Java

2.Android Studio

3. Windows 10 OS

2.1.2 Secondary Requirement

Secondary requirements are future considerations that can be incorporated into the application for further development:

· Ability to integrate history page in the application

· Ability to receive notifications through the Android Notification service when new products are available on the server

· Ability for the user to suggest product(s) of interest

· Ability for the user to be able to read some portion of the product

· Ability for the user to comment on, recommend and rate an item.

2.2 Algorithm Flow Chart

A successful and dynamic application should have a flow of events that implements the system behaviour. The term algorithm describes a solution to a problem. Algorithm is important in a problem-solving environment because it states the steps and procedures leading to the solution. In other words, an algorithm provides step-by-step procedures in solving a particular problem. On the other hand, a flowchart is a diagram representing the flow of a process in a system.

It combines symbols and flow lines, to visually represent the operation of the algorithm.

The Algorithm Flowchart therefore is a figurative representation of the entire process,

describing a set of instructions executed step-by-step to solve a given problem.

When the application is launched, the user has the option of either registering (for a complete transaction) or skipping the registration process (for quick browsing of product(s) without making a purchase). If the user choses to register, the system displays a registration form to be completed by the user. After this registration process, the user can log in with the login credentials in order to search for products, select desired product(s), add product(s) to the cart and purchase product(s).



* 1. **Use Case**

A use case describes how a system behaves and responds to inputs by the primary actors. The primary actors interact with the system in a bid to accomplish a desired goal. The system, on the other hand, responds to the actors with the expected results. Fig describes the use case of the Online Productstore system. The use case brings together all possible scenarios under which a given system can act or behave.

In this project, the actors are the CLIENT, VISITOR, MERCHANT and the PAYUMONEY service. The use cases are REGISTER, BROWSE HOME PAGE, UPDATE SHOPPING CART, LOGIN, PLACE ORDER, SHIP ORDER, MANAGE SHOPPING CART, CREDIT MERCHANT ACCOUNT, MANAGE PRODUCT CATALOG, LOGOUT, ADD

ITEM TO CART, SEARCH ITEMS, and PERFORM ITEM SORTING.

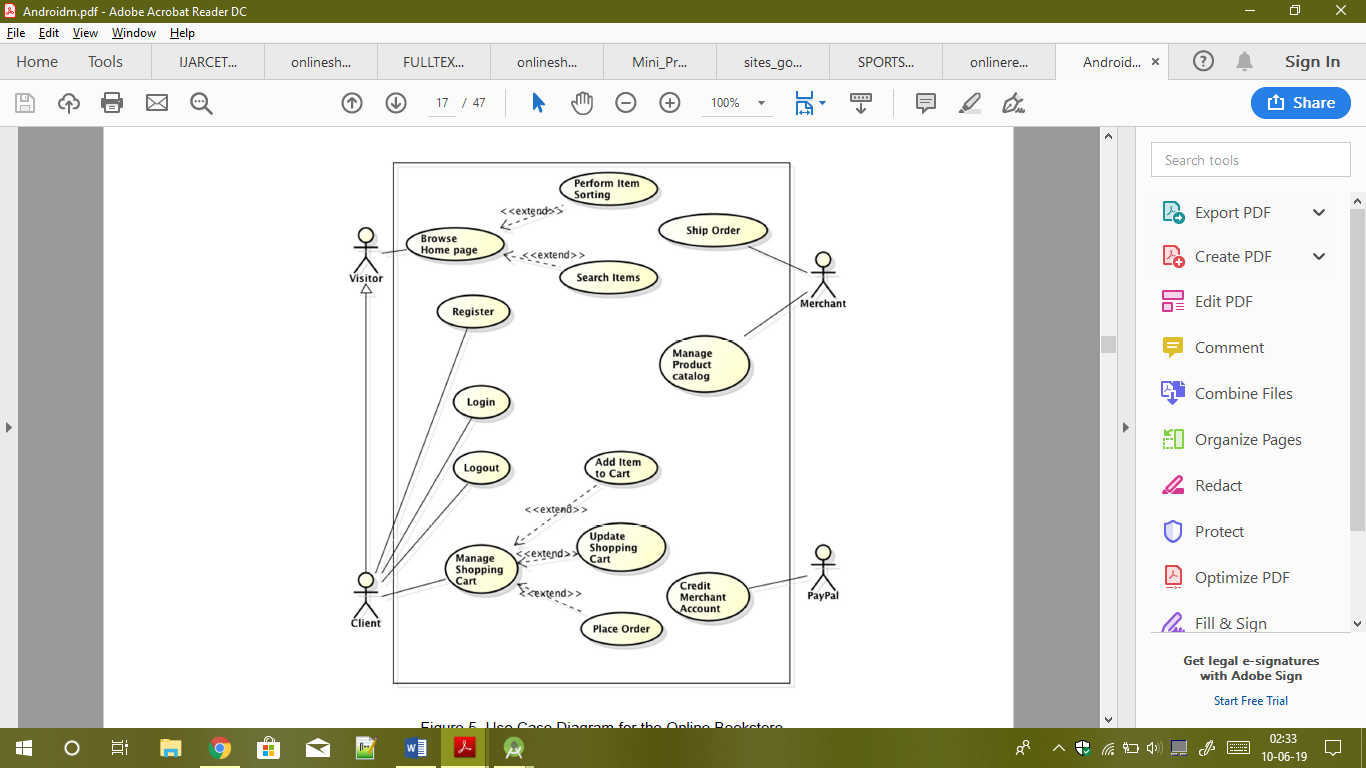
The actors of the system are described below:

CLIENT: The client is a registered and authorised user also known as a customer. The client is able to log in to the system to purchase items from the online store.

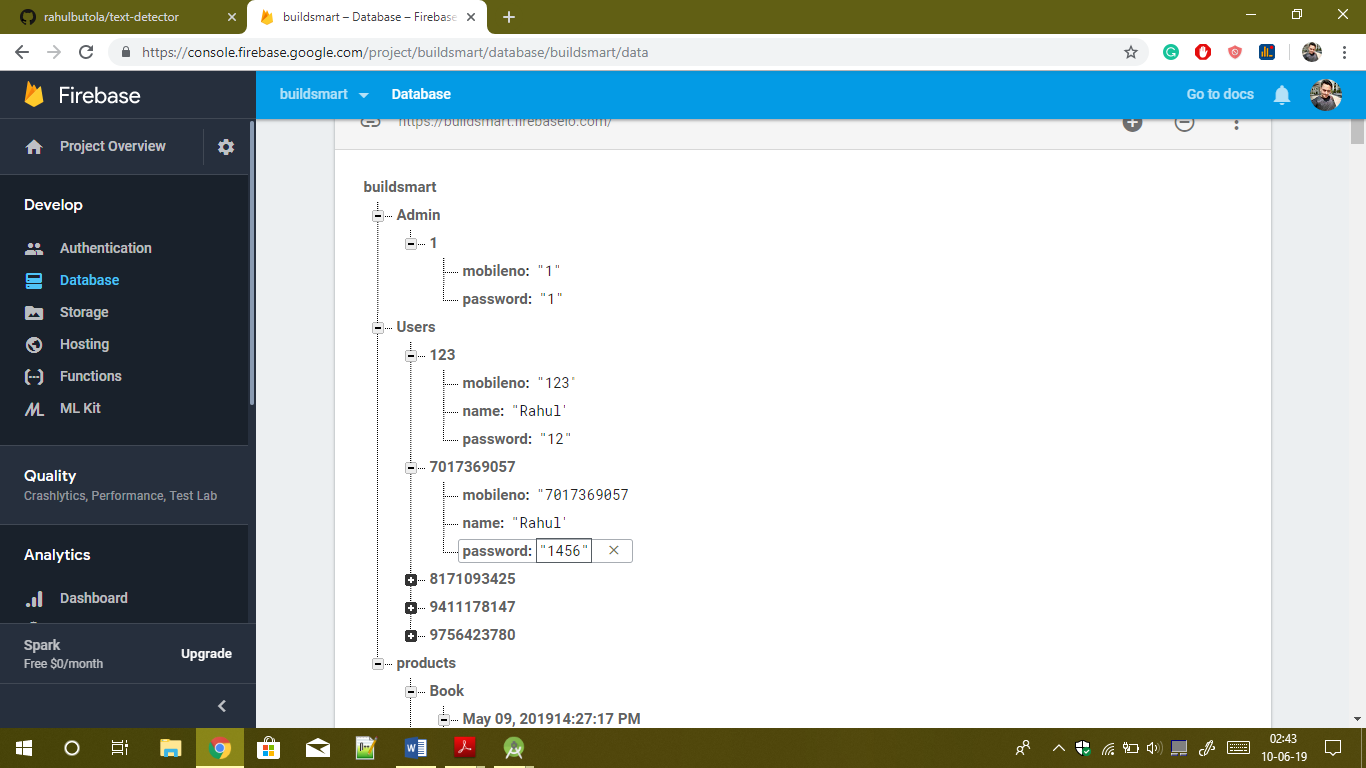
VISITOR: The visitor is unregistered or an anonymous user that is able to search for and view items from the online store system. The visitor cannot log onto the system and make a purchase unless he/she is registered.

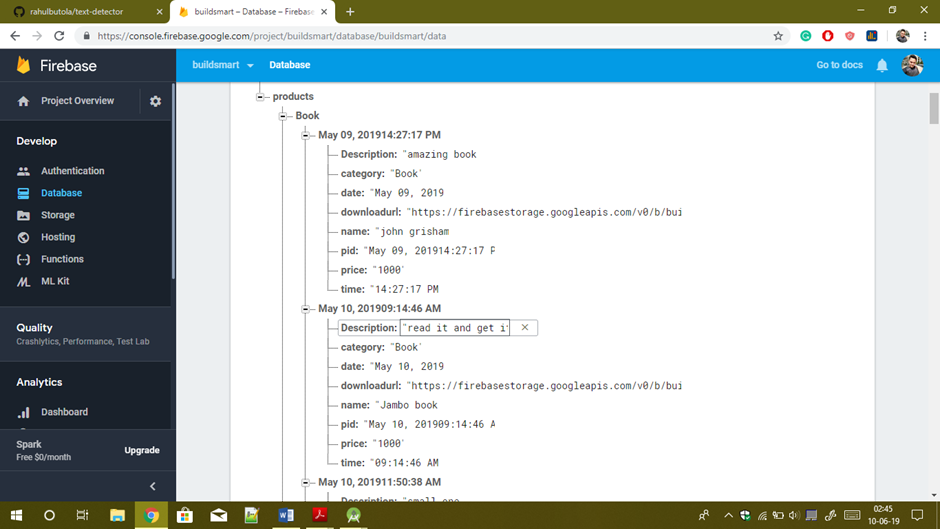
MERCHANT: The merchant represents a person selling the product through the online system, populates the product catalog and updates any information related to the products in the system. Also, the merchant ships orders to the client.

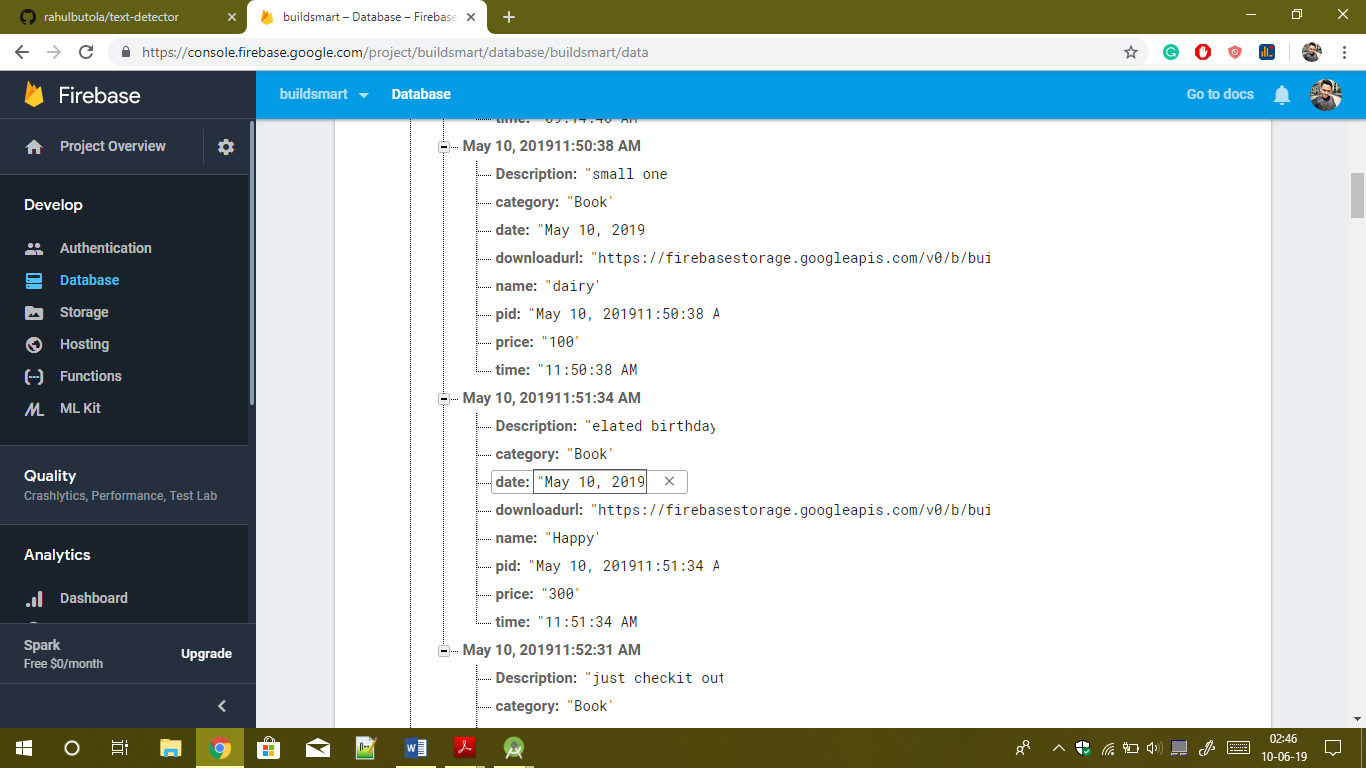
PAYUMONEY: The PayUMoney is responsible for validating a client’s credit card information, debits the client’s account and credits the merchant’s account during transaction.

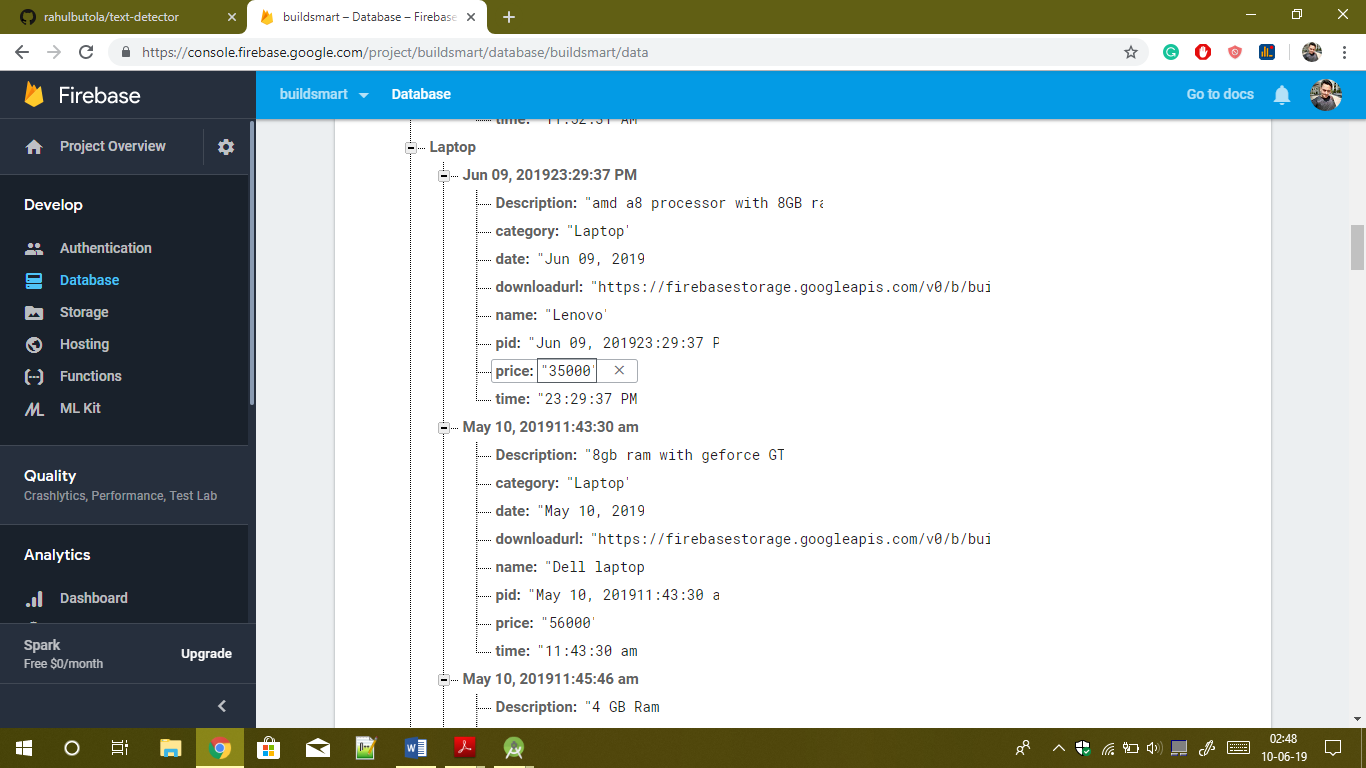


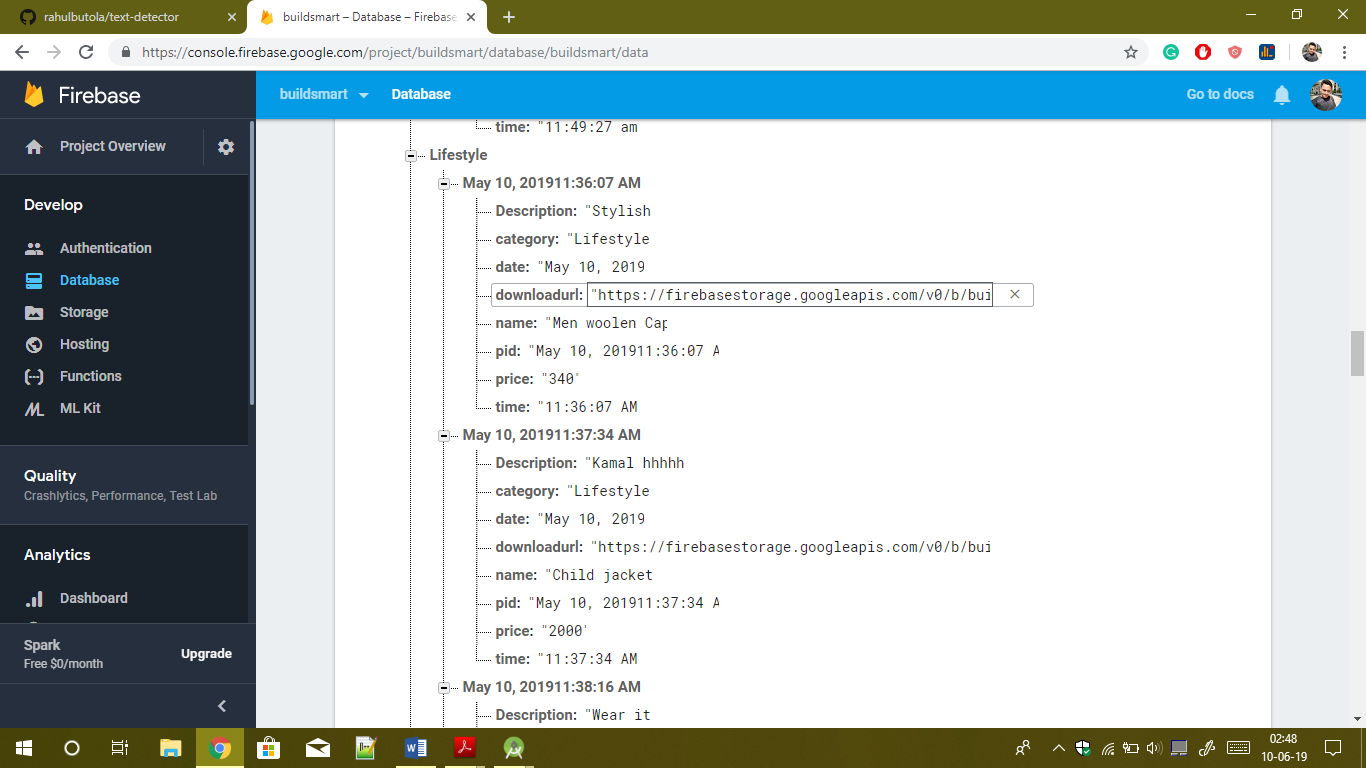
* 1. **Datbase Structure**

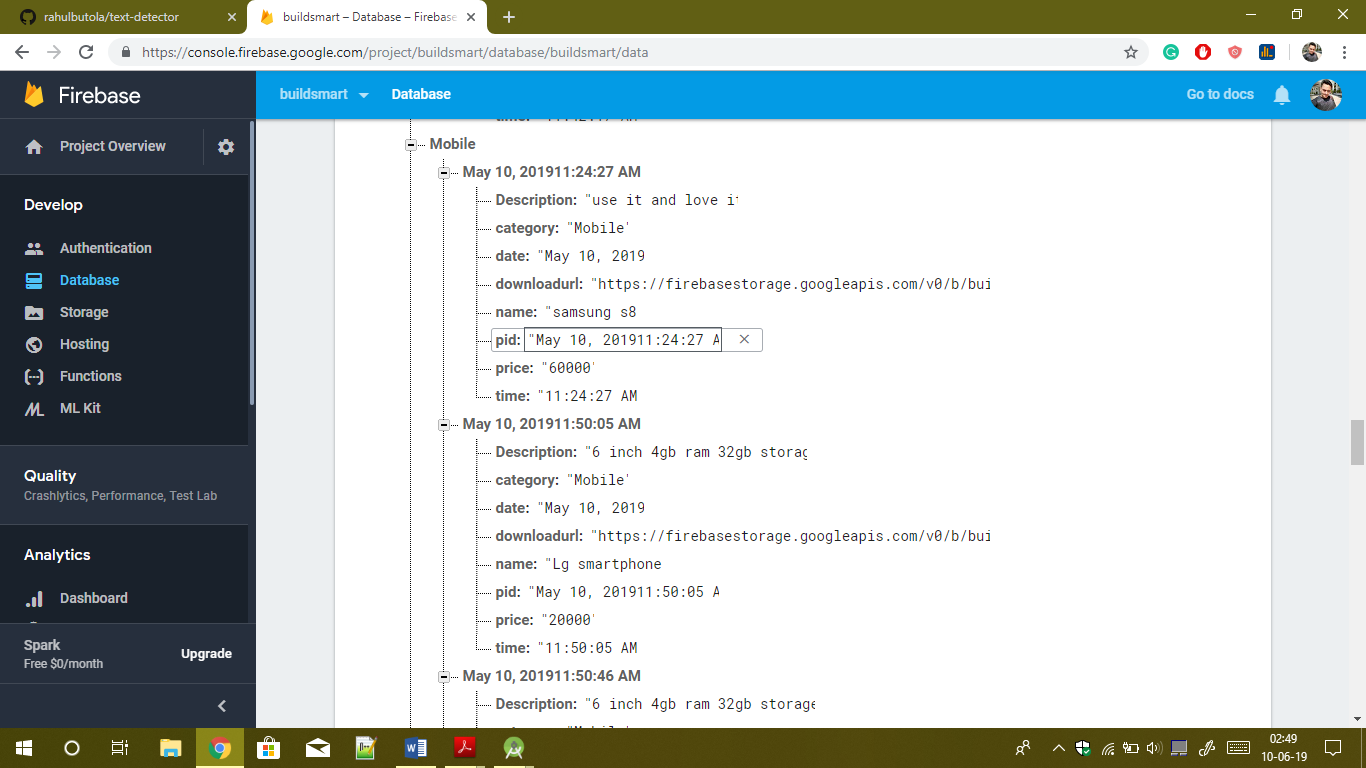


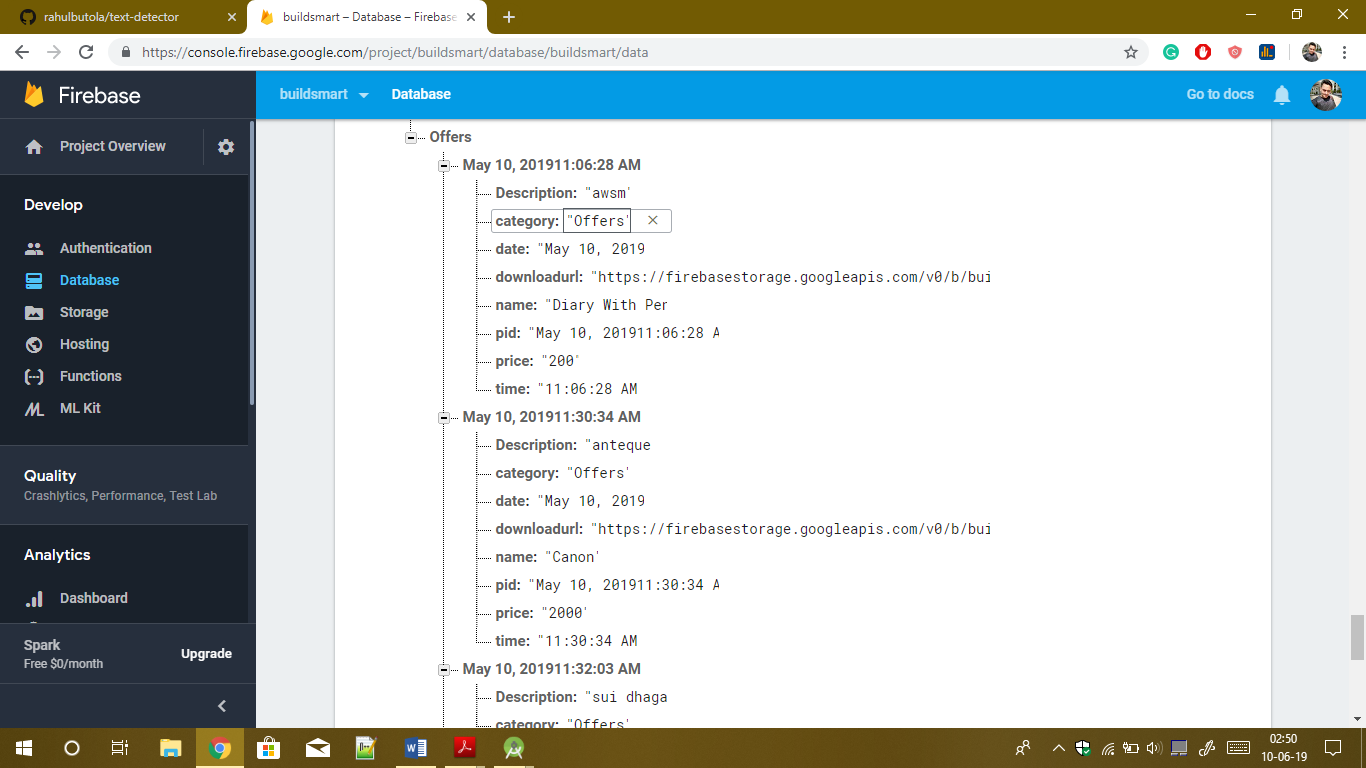












**2.5 Modules**

2.5.1 Administrator Module:

Admin has all privileges to add/modify cashier details In this there are sub modules like

1. Add New Product
2. Login

2.5.1.1 Add New Product:

This module only for product maintenance ading new product details in the store.

2.5.2 Customer Module:

2.5.2.1. REGISTER: A new user that wants to purchase a product must register into the database prior to a transaction. The application displays the registration form to the user.

2.5.2.2 LOGIN: A registered user that wants to buy product(s) needs to LOGIN before any transaction

can take place.

2.5.2.5 SEARCH ITEMS: The SEARCH ITEMS use case extends BROWSE HOME PAGE and allows the user to search the Product and Promotions pages for a particular item, browse through items and perform sorting of the items. The user can find an item quickly by the name or title of the product using the search button on these pages. The

skip button on the application makes it possible for an unregistered user to search through the collection of products.

2.5.2.4 ADD ITEM TO CART: The ADD ITEM TO CART use case extends MANAGE SHOPPING CART and provides the user with the opportunity to place an order. The user must be registered and successfully logged in before adding items to Cart.

2.5.2.5 PLACE ORDER: The PLACE ORDER use case describes how the client completes a purchase by checking out from the shopping cart. It provides the user with the opportunity to make a payment for an order. Thus, a user must be registered before making a payment using the PayPal paying system.

2.5.2.6 BROWSE HOME PAGE: The BROWSE HOME PAGE use case allows both the visitor and the client to view a list of products on the product page, which is stored in a relational database. Before this can be performed, a database populated with a list of products must exist. User first requests a list of products in the database, then the web server connects to the database and displays all available products on the product page.

2.5.2.7 UPDATE SHOPPING CART: The UPDATE SHOPPING CART use case allows the client to update the shopping cart. The client has a user account and must have logged into the system. The client can remove item(s), add item(s) or empty the cart.

2.5.2.8 MANAGE SHOPPING CART: The shopping cart is implemented on the application. This use case describes the activities the client can perform on the shopping cart. These include browsing the product page, viewing the cart, adding to or removing items from the cart and editing the quantities of items in the cart.

2.5.2.9 CREDIT MERCHANT ACCOUNT: The CREDIT MERCHANT ACCOUNT use case is utilised by the actor PAYUMONEY in order to credit the merchant account when a client has successfully placed an order. The PayPal service validates the client’s credit card information

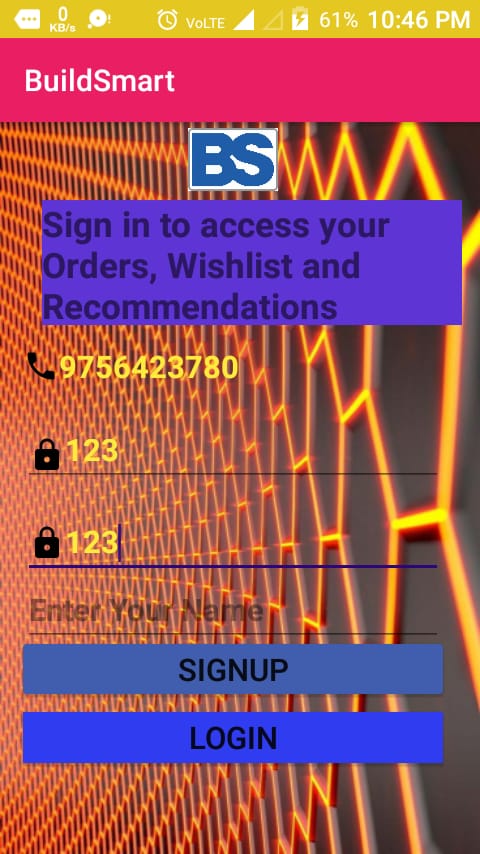
before crediting the merchant account. The system displays the transaction details.

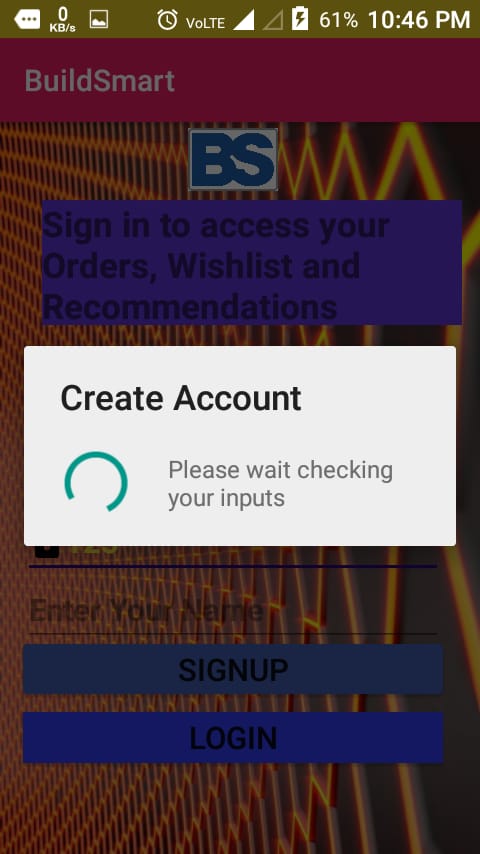
2.5.2.10 MANAGE PRODUCT CATALOG: The MANAGE PRODUCT CATALOG use case allows the merchant to manage the products page. The merchant can add to, remove or update product prices in the catalog. The merchant can also manage the inventory and sales.

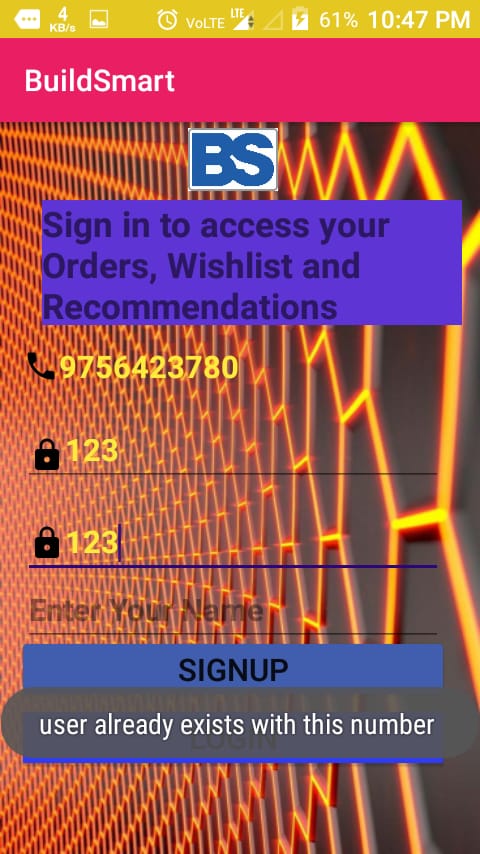
2.5.2.11 LOGOUT: The LOGOUT use case requires that a client has been log into the system. The client is able to click logout after a successful transaction or at any point of the process to interrupt the system. The system displays a prompt to confirm if a user intends to log out. If the client selects logout, the system returns the client to the login page and ends the session. Otherwise, the client continues from the last page before clicking logout.

**CHAPTER** **3**

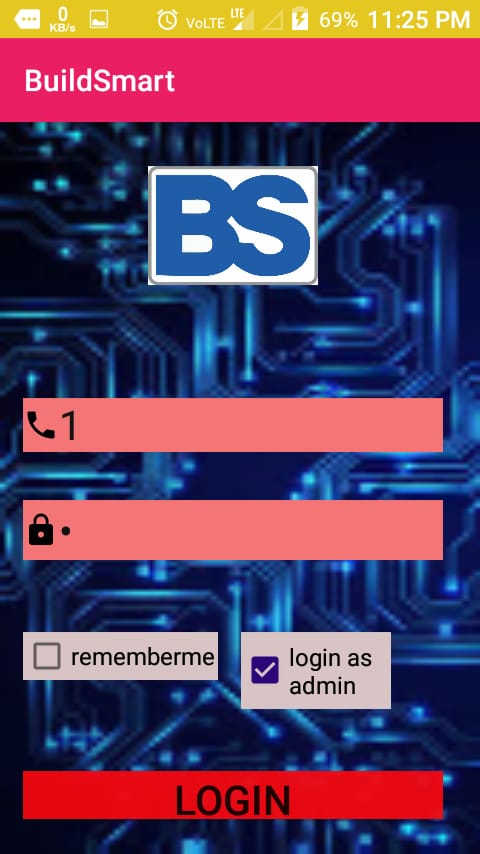
**SNAPSHOT OF PROJECT**

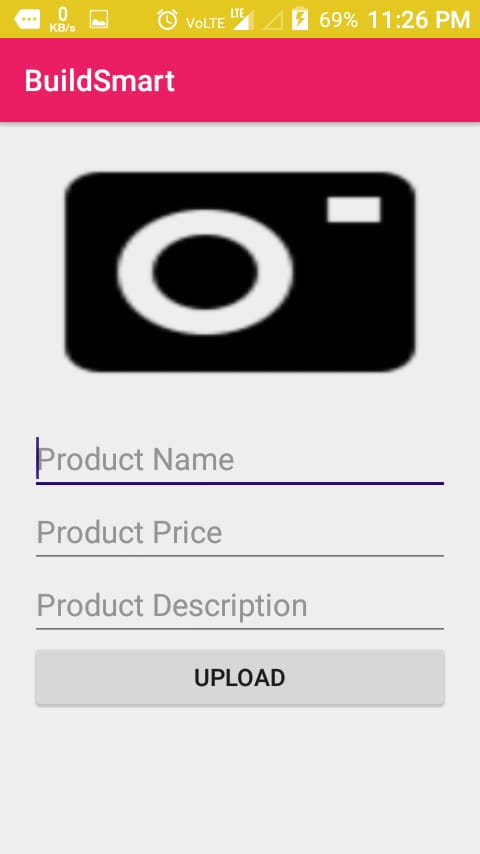
**3.1 SignUp page**

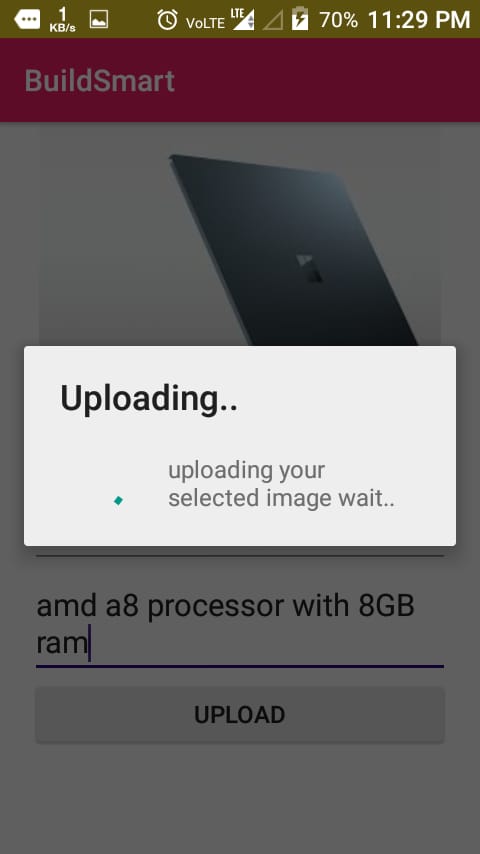
****

****

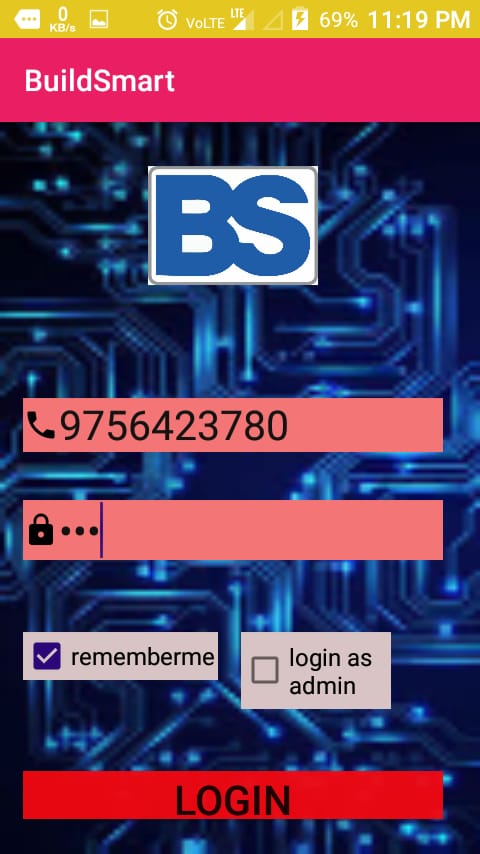
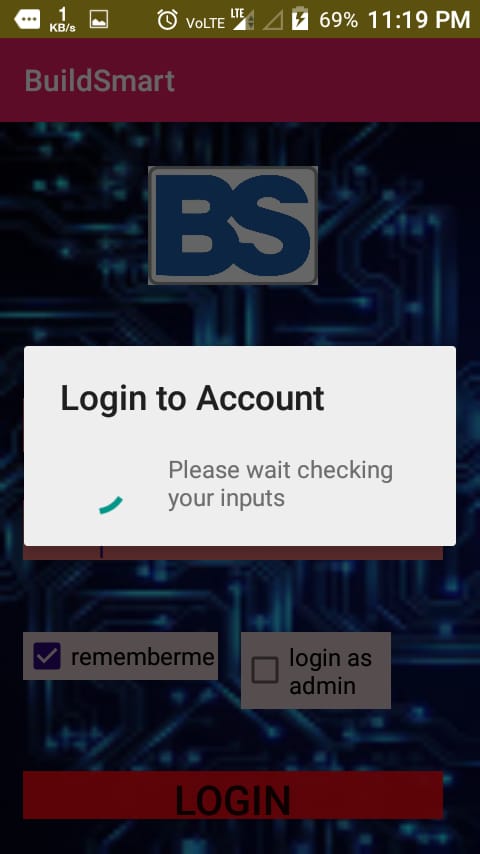
**3.2 Login by Admin**

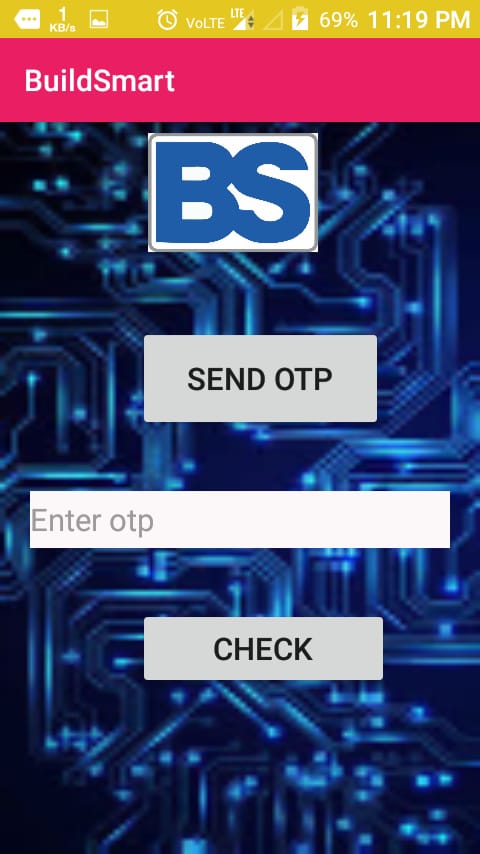
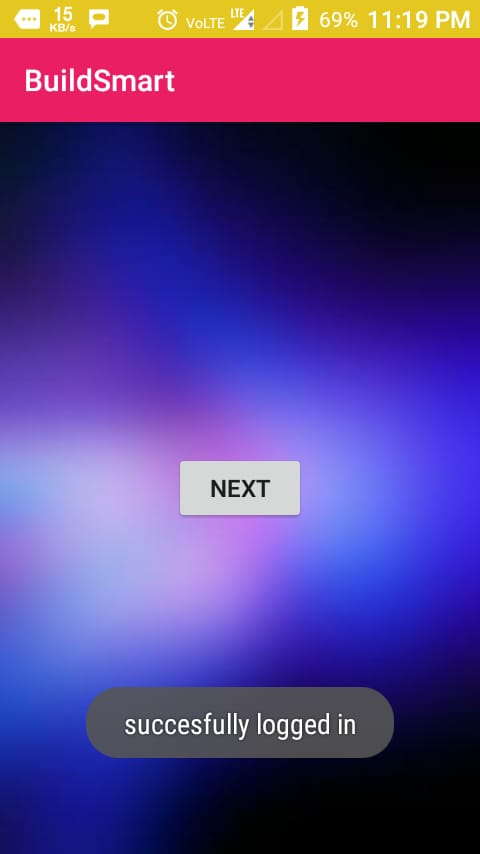
 

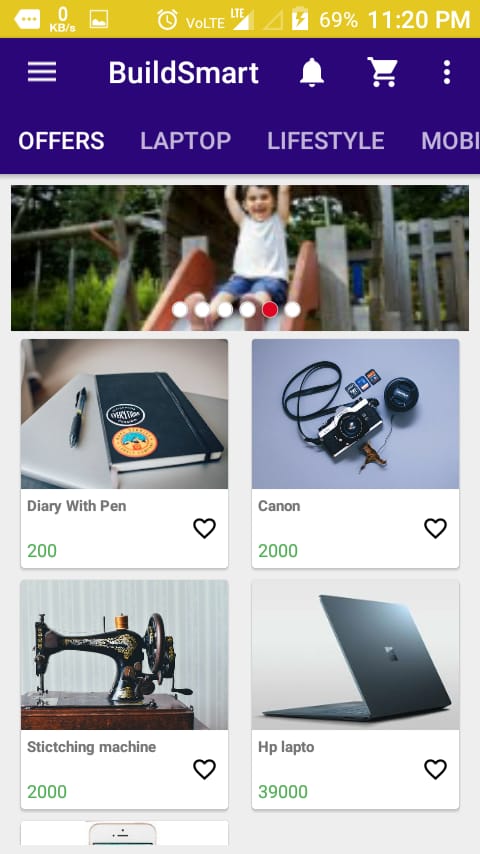
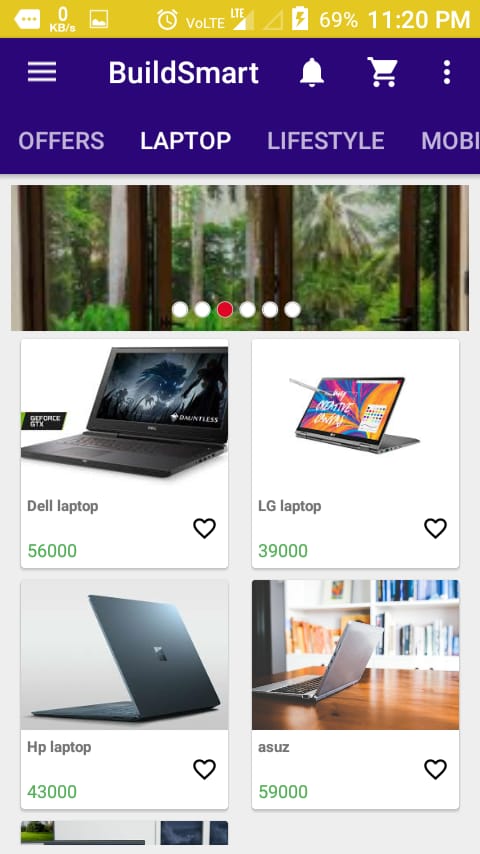
**3.3 Login by User**

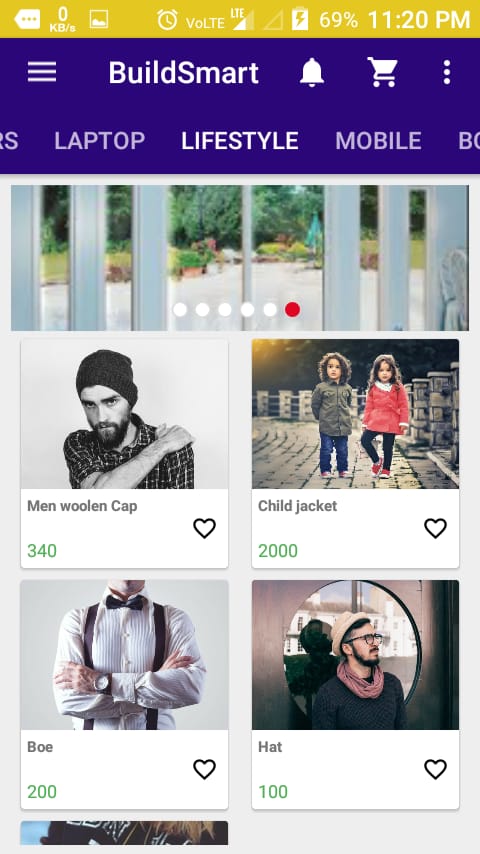
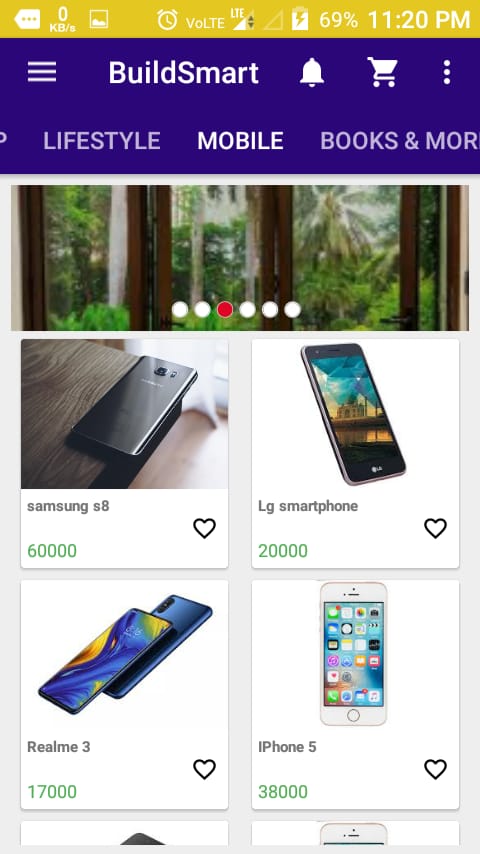
** **

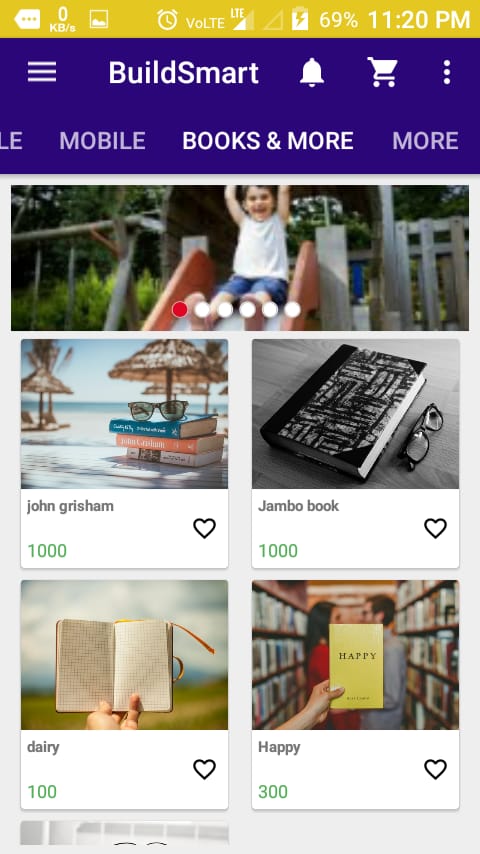
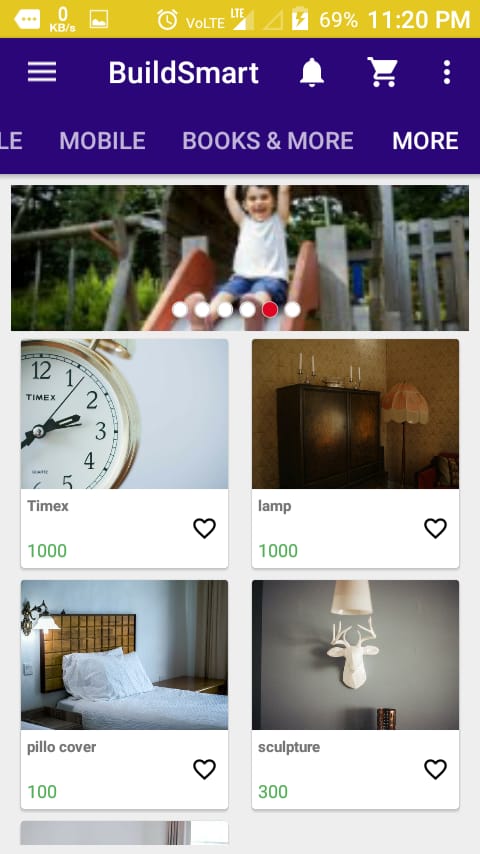
**3.4 OTP Generation**

** **

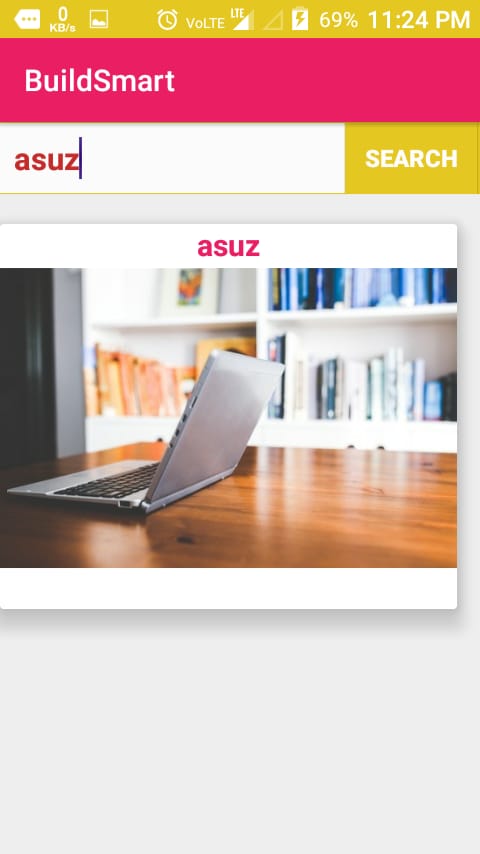
**3.4 Home Page Display**

** **

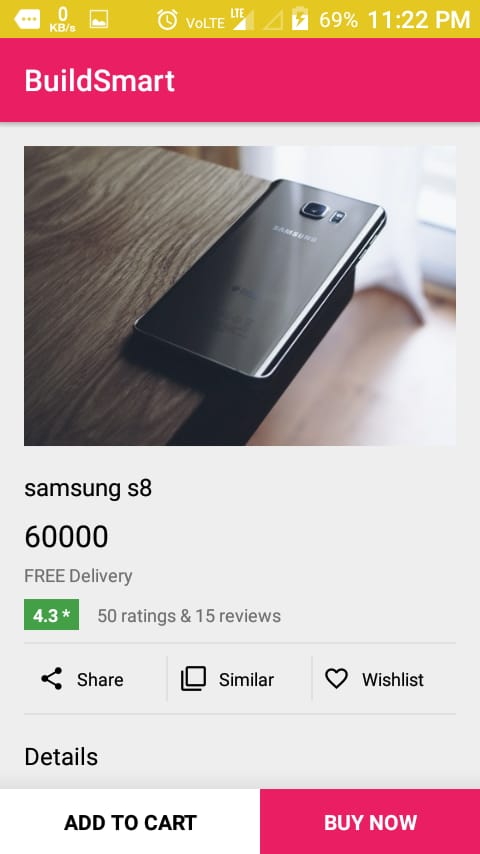
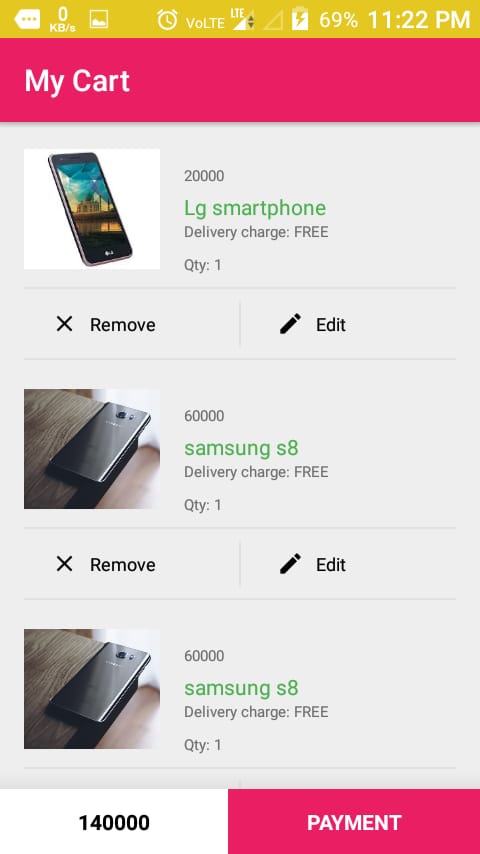
** **

** **

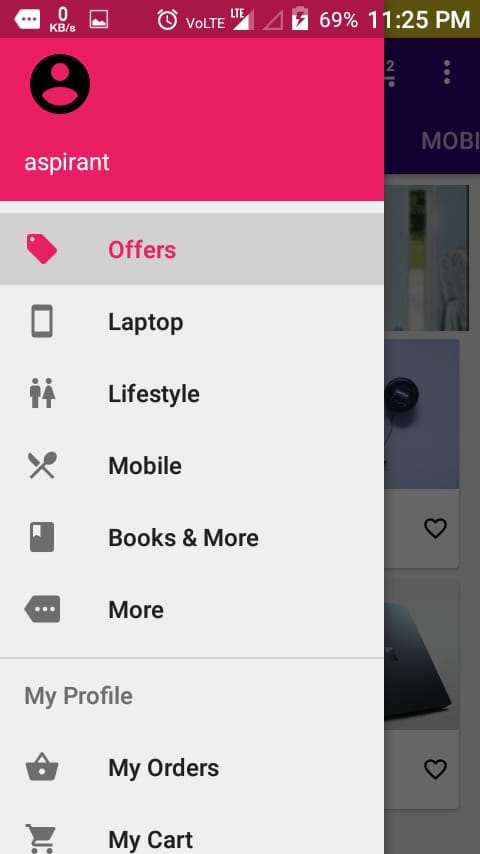
**3.5 search**



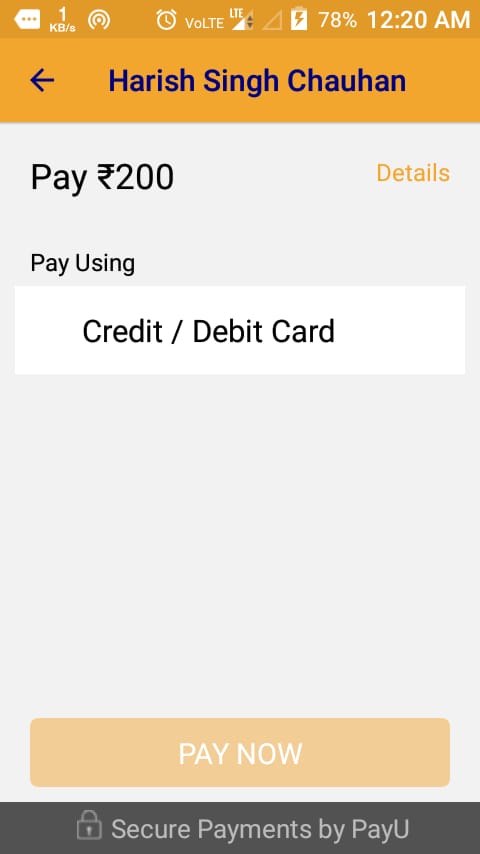
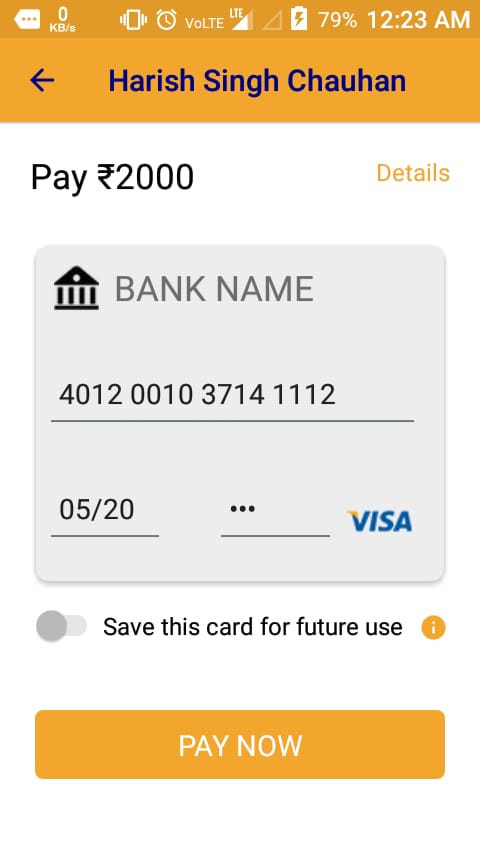
**3.6 Add to cart**

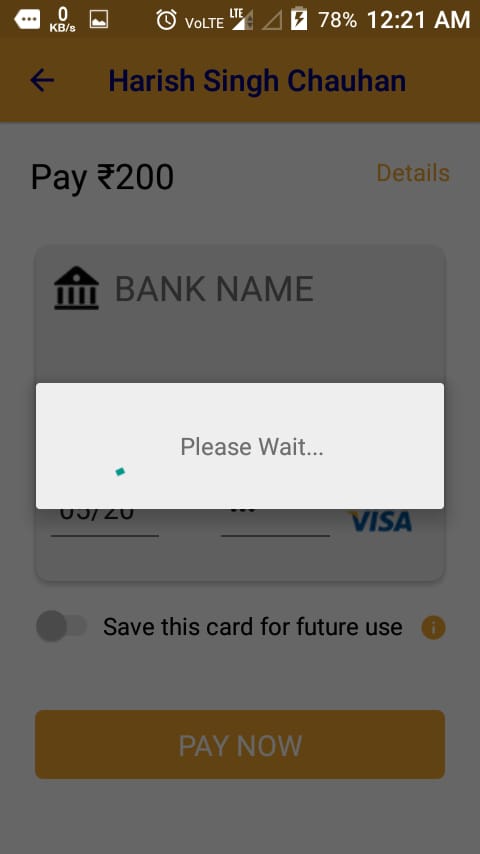
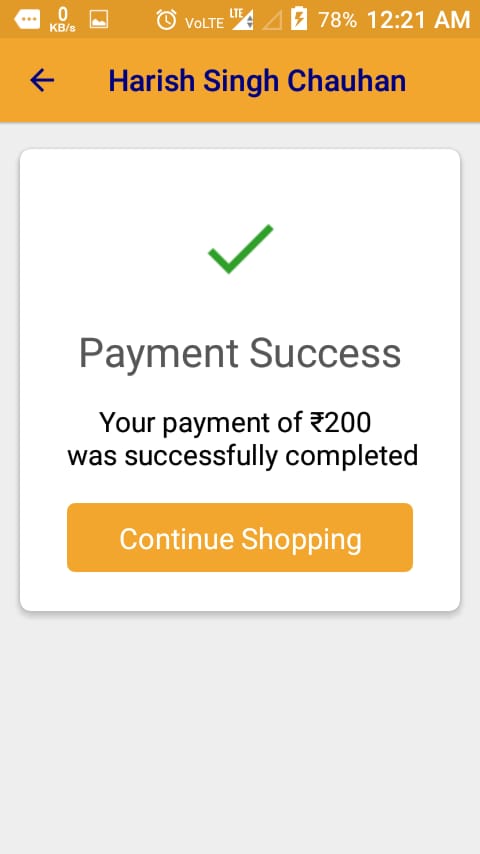
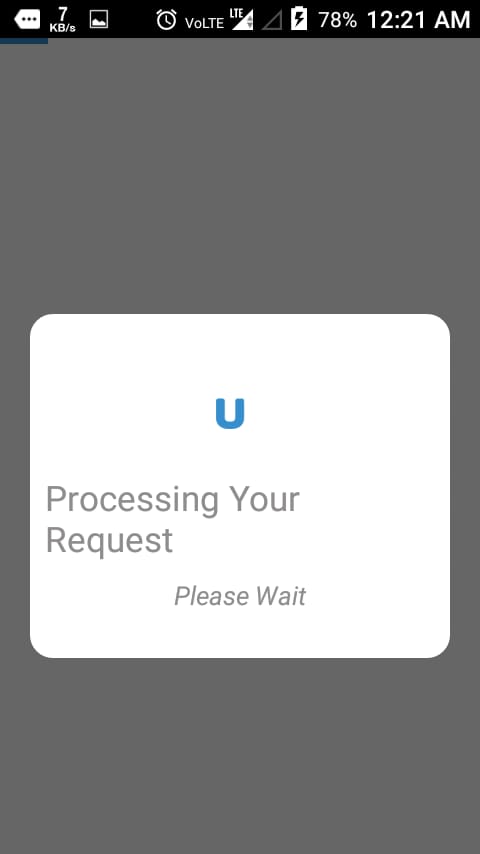
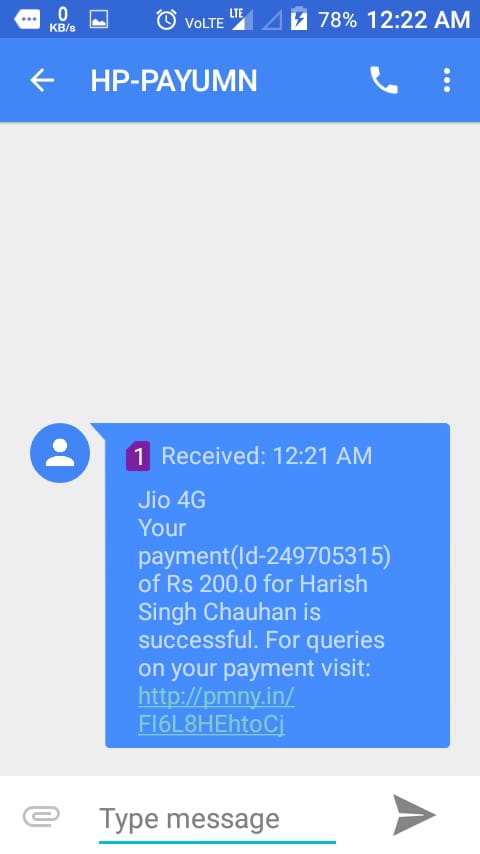
** **

**3.7 Drawer**



**3.7 Payment**

** **

**  **

**CHAPTER 4**

**CONCLUSION**

The project was carried out to develop an Android phone application for an online ecommerce system with PayUMoney Integration. The goal of the project is to create an Android application that would allow users to register an account, login, search for particular thing of interest purchase

thing(s) in the cart with a PayUMoney account.

The objectives of the project were achieved by observing software development procedures and principles for software designs and implementation. In achieving the goal of this project, three major parts were designed and implemented. Firstly, the design of the UI is attractive, intuitive, responsive and with good user experience in mind. This was achieved and implemented by following the Android design guidelines for Android devices.

Furthermore, all required functionalities were implemented accordingly and, hence, a fully operative and functional Android phone application was developed. The application is able to register an account for users, while protecting and encrypting a user’s password. The users can log in conveniently with the username and password, search

for available items on the online store, order things and make purchases using the integrated PayUMoney services. In addition, users can log out from the online store.

4.2 FUTURE WORKS

In conclusion, it is important to know that this application could still be improved upon by adding more interesting features. For example, a feature that makes the user to save the purchase history could be added. In addition, the application could demonstrate a feature that allows the user to not only rate things, but also recommend

product(s) of interest to friends.

**APPENDIX**

**Code**

**1.AndroidManifest.xml**

<?xml version="1.0" encoding="utf-8"?>

<manifest package="com.allandroidprojects.buildsmart"

xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools">

<uses-permission android:name="android.permission.INTERNET"/>

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"/>

<uses-permission android:name="android.permission.SEND\_SMS"/>

<uses-permission android:name="android.permission.GET\_ACCOUNTS" />

<uses-permission android:name="android.permission.READ\_PROFILE" />

<uses-permission android:name="android.permission.READ\_CONTACTS" />

<uses-permission android:name="android.permission.READ\_PHONE\_STATE"/>

<supports-screens

android:anyDensity="true"

android:largeScreens="true"

android:smallScreens="true"

android:normalScreens="true"

android:xlargeScreens="true"

android:resizeable="true"/>

<application

android:name="com.allandroidprojects.buildsmart.startup.FrescoApplication"

android:allowBackup="true"

android:icon="@mipmap/ic\_launcher\_bs"

android:roundIcon="@mipmap/ic\_launcher\_bs"

android:label="@string/app\_name"

android:supportsRtl="true"

android:theme="@style/AppTheme"

tools:replace="android:allowBackup">

<activity android:name="com.allandroidprojects.buildsmart.startup.SplashActivity">

<intent-filter>

<action android:name="android.intent.action.MAIN"/>

<category android:name="android.intent.category.LAUNCHER"/>

</intent-filter>

<meta-data

android:name="android.app.default\_searchable"

android:value=".options.SearchResultsActivity"/>

</activity>

<activity android:name="com.allandroidprojects.buildsmart.LoginAndsignup.SignUp" />

<activity android:name="com.allandroidprojects.buildsmart.LoginAndsignup.Login" />

<activity android:name="com.allandroidprojects.buildsmart.LoginAndsignup.OTP"></activity>

<activity

android:name="com.allandroidprojects.buildsmart.startup.WelcomeActivity"

android:theme="@style/AppTheme.NoActionBar">

</activity>

<activity

android:name="com.allandroidprojects.buildsmart.startup.MainActivity"

android:label="@string/app\_name"

android:theme="@style/AppTheme.NoActionBar">

</activity>

<activity android:name="com.allandroidprojects.buildsmart.fragments.ViewPagerActivity"/>

<activity android:name="com.allandroidprojects.buildsmart.product.ItemDetailsActivity">

</activity>

<activity

android:name="com.allandroidprojects.buildsmart.options.WishlistActivity"

android:label="@string/my\_wishlist">

</activity>

<activity

android:name="com.allandroidprojects.buildsmart.options.CartListActivity"

android:label="@string/my\_cart">

</activity>

<activity android:name="com.allandroidprojects.buildsmart.options.SearchResultActivity">

<!-- to identify this activity as "searchable.xml" -->

<intent-filter>

<action android:name="android.intent.action.SEARCH"/>

<category android:name="android.intent.category.DEFAULT"/>

</intent-filter>

<meta-data

android:name="android.app.searchable"

android:resource="@xml/searchable"/>

</activity>

<activity android:name="com.allandroidprojects.buildsmart.miscellaneous.EmptyActivity">

</activity>

<activity android:name="com.allandroidprojects.buildsmart.payu\_de.MainActivity2">

</activity>

<activity android:name="com.allandroidprojects.buildsmart.payu\_de.StartPaymentActivity"/>

<activity android:name=".Category.AddCategory" />

<activity android:name=".Category.UploadCategoryItem" />

<activity android:name=".DisplayHome.HomeActivity"></activity>

</application>

</manifest>

2. LoginAndsignup

2.a Login

**package** com.allandroidprojects.buildsmart.LoginAndsignup;  
  
**import** android.app.ProgressDialog;  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.text.TextUtils;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.CheckBox;  
**import** android.widget.EditText;  
**import** android.widget.Toast;  
  
**import** com.allandroidprojects.buildsmart.Category.AddCategory;  
**import** com.allandroidprojects.buildsmart.DisplayHome.HomeActivity;  
**import** com.allandroidprojects.buildsmart.R;  
**import** com.google.firebase.database.DataSnapshot;  
**import** com.google.firebase.database.DatabaseError;  
**import** com.google.firebase.database.DatabaseReference;  
**import** com.google.firebase.database.FirebaseDatabase;  
**import** com.google.firebase.database.ValueEventListener;  
  
**public class** Login **extends** AppCompatActivity {  
  
 EditText **mobileno**,**password**;  
 Button **loginAcc**;  
 DatabaseReference **databaseReference**;  
 ProgressDialog **lodingBar**;  
 String **user**;  
 CheckBox **asAdmin**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_login***);  
 *// databaseReference.initializeApp(this);* **mobileno**=findViewById(R.id.***mobileno2***);  
 **password**=findViewById(R.id.***password2***);  
 **loginAcc**=findViewById(R.id.***loginacc1***);  
 **lodingBar**=**new** ProgressDialog(**this**);  
 **asAdmin**=findViewById(R.id.***checkbox\_admin***);  
 **loginAcc**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
  
 String no=**mobileno**.getText().toString();  
 String pass=**password**.getText().toString();  
  
 **if**(TextUtils.*isEmpty*(no))  
 {  
 Toast.*makeText*(getApplicationContext(),**"mobileno can't be empty"**,Toast.***LENGTH\_SHORT***).show();  
 }  
 **else if**(TextUtils.*isEmpty*(pass))  
 {  
 Toast.*makeText*(getApplicationContext(),**"password can't be empty"**,Toast.***LENGTH\_SHORT***).show();  
 }  
  
 **else** {  
 **lodingBar**.setTitle(**"Login to Account"**);  
 **lodingBar**.setMessage(**"Please wait checking your inputs"**);  
 **lodingBar**.setCanceledOnTouchOutside(**false**);  
 **lodingBar**.show();  
  
 validate(no,pass);  
 }  
  
 }  
 });  
  
  
 }  
 **public void** validate(**final** String no, **final** String pass) {  
 **if**(**asAdmin**.isChecked())  
 { **user**=**"Admin"**;  
 }  
 **else** { **user**=**"Users"**;}  
 DatabaseReference dref = FirebaseDatabase.*getInstance*().getReference();  
  
 dref.addListenerForSingleValueEvent(**new** ValueEventListener() {  
 @Override  
 **public void** onDataChange(DataSnapshot dataSnapshot) {  
 **if** (dataSnapshot.child(**user**).child(no).exists()) {  
  
 Model\_Users userData = dataSnapshot.child(**user**).child(no).getValue(Model\_Users.**class**);  
  
  
 **if** (no.equals(userData.getMobileno())) {  
 **if** (pass.equals(userData.getPassword())) {  
 Toast.*makeText*(getApplicationContext(),  
 **"Successfully logged in"**, Toast.***LENGTH\_SHORT***).show();  
 Intent i;  
 **if**(**asAdmin**.isChecked()) {  
 i = **new** Intent(Login.**this**, AddCategory.**class**);  
 }  
 **else** {  
 i = **new** Intent(Login.**this**, OTP.**class**);  
 }  
 i.putExtra(**"no"**,no);  
 startActivity(i);  
 **lodingBar**.dismiss();  
 } **else** {  
 Toast.*makeText*(getApplicationContext(),  
 **"Wrong password try again"**, Toast.***LENGTH\_SHORT***).show();  
 **lodingBar**.dismiss();  
 }  
  
 }  
 **else** {  
 Toast.*makeText*(getApplicationContext(),  
 **"wrong number try again"**, Toast.***LENGTH\_SHORT***).show();  
 **lodingBar**.dismiss();  
 }  
 }  
 **else** {  
 Toast.*makeText*(getApplicationContext(),  
 **"user doesnot exist"**, Toast.***LENGTH\_SHORT***).show();  
  
 **lodingBar**.dismiss();  
  
 }  
 }  
  
  
 @Override  
 **public void** onCancelled(DatabaseError databaseError) {  
  
 }  
 });  
  
  
  
  
 }  
}

2.b Model\_Users

**package** com.allandroidprojects.buildsmart.LoginAndsignup;  
  
**public class** Model\_Users {  
 String **mobileno**,**password**;  
  
 **public** Model\_Users() {  
 }  
  
 **public** Model\_Users(String mobileno , String password) {  
 **this**.**mobileno** = mobileno;  
 **this**.**password**= password;  
 }  
  
 **public** String getMobileno() {  
 **return mobileno**;  
 }  
  
 **public void** setMobileno(String mobileno) {  
 **this**.**mobileno** = mobileno;  
 }  
  
 **public** String getPassword() {  
 **return password**;  
 }  
  
 **public void** setPassword(String password) {  
 **this**.**password** = password;  
 }  
}

2.c OTP

**package** com.allandroidprojects.buildsmart.LoginAndsignup;  
  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.telephony.SmsManager;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.Toast;  
  
**import** com.allandroidprojects.buildsmart.DisplayHome.HomeActivity;  
**import** com.allandroidprojects.buildsmart.R;  
**import** com.allandroidprojects.buildsmart.startup.MainActivity;  
  
**import** java.util.Random;  
  
**public class** OTP **extends** AppCompatActivity {  
Button **send**,**check**;  
EditText **otp**;  
SmsManager **sm**;  
String **s**;  
**int rand**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_otp***);  
 **send**=findViewById(R.id.***button2***);  
 **check**=findViewById(R.id.***button3***);  
 **sm**= SmsManager.*getDefault*();  
 Intent i= getIntent();  
 Random random= **new** Random();  
 **rand**= random.nextInt(50000);  
 **otp**=findViewById(R.id.***editText***);  
 **s**=i.getStringExtra(**"no"**);  
 **send**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 **sm**.sendTextMessage(**s**,**null**,**"your OTP is : "**+**rand**,**null**,**null**);  
  
 }  
 });  
 **check**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 **int** temp=Integer.*parseInt*(**otp**.getText().toString());  
 **if**(temp==**rand**)  
 {  
  
 Intent i=**new** Intent(getApplicationContext(), HomeActivity.**class**);  
 i.putExtra(**"phnno"**,**s**);  
 startActivity(i);  
 Toast.*makeText*(getApplicationContext(),**"succesfully logged in"**,Toast.***LENGTH\_SHORT***).show();  
 }  
 **else** {  
 Toast.*makeText*(getApplicationContext(),**"wrong otp"**,Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 }); }  
}

2.d Signup

**package** com.allandroidprojects.buildsmart.LoginAndsignup;  
  
**import** android.app.ProgressDialog;  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.support.annotation.NonNull;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.text.TextUtils;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.Toast;  
**import** com.allandroidprojects.buildsmart.R;  
**import** com.google.android.gms.tasks.OnCompleteListener;  
**import** com.google.android.gms.tasks.Task;  
**import** com.google.firebase.database.DataSnapshot;  
**import** com.google.firebase.database.DatabaseError;  
**import** com.google.firebase.database.DatabaseReference;  
**import** com.google.firebase.database.FirebaseDatabase;  
**import** com.google.firebase.database.ValueEventListener;  
**import** java.util.HashMap;  
**public class** SignUp **extends** AppCompatActivity {  
EditText **mobileno**,**password**,**name**;  
Button **createAcc**,**login**;  
String **sr**;  
DatabaseReference **databaseReference**;  
ProgressDialog **lodingBar**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***sign\_up***);  
 **mobileno**=findViewById(R.id.***mobileno1***);  
 **password**=findViewById(R.id.***password1***);  
  
 **createAcc**=findViewById(R.id.***createacc1***);  
 **lodingBar**=**new** ProgressDialog(**this**);  
 **login**=findViewById(R.id.***login2***);  
  
 **login**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 Intent i=**new** Intent(SignUp.**this**,Login.**class**);  
 startActivity(i);  
 }  
 });  
  
 **createAcc**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 **if**(TextUtils.*isEmpty*(**mobileno**.getText().toString()))  
 {  
 Toast.*makeText*(getApplicationContext(),**"mobileno can't be empty"**,Toast.***LENGTH\_SHORT***).show();  
 }  
 **else if**(TextUtils.*isEmpty*(**password**.getText().toString()))  
 {  
 Toast.*makeText*(getApplicationContext(),**"password can't be empty"**,Toast.***LENGTH\_SHORT***).show();  
 }  
 **else** {  
 **lodingBar**.setTitle(**"Create Account"**);  
 **lodingBar**.setMessage(**"Please wait checking your inputs"**);  
 **lodingBar**.setCanceledOnTouchOutside(**false**);  
 **lodingBar**.show();  
 createAccount(**mobileno**.getText().toString(),**password**.getText().toString());  
 }  
 }  
 });  
  
 }  
 **protected void** createAccount(**final** String mobileno1, **final** String password1)  
 {  
 **databaseReference**= FirebaseDatabase.*getInstance*().getReference();  
 **final** String ph=mobileno1;  
  
 **name**=findViewById(R.id.***editText3***);  
 **sr**=**name**.getText().toString();  
  
 **databaseReference**.addListenerForSingleValueEvent(**new** ValueEventListener() {  
 @Override  
 **public void** onDataChange(DataSnapshot dataSnapshot) {  
  
 **if**(!(dataSnapshot.child(**"Users"**).child(ph).exists()))  
 {  
 HashMap<String,Object> usedataMap=**new** HashMap<>();  
 usedataMap.put(**"mobileno"**,mobileno1);  
 usedataMap.put(**"password"**,password1);  
 usedataMap.put(**"name"**,**sr**);  
  
 **databaseReference**.child(**"Users"**).child(mobileno1).updateChildren(usedataMap)  
 .addOnCompleteListener(**new** OnCompleteListener<Void>() {  
 @Override  
 **public void** onComplete(@NonNull Task<Void> task) {  
 **if**(task.isSuccessful())  
 {Toast.*makeText*(getApplicationContext(),**"acc created successfully"**,Toast.***LENGTH\_SHORT***).show();  
 **lodingBar**.dismiss();  
  
 }  
 **else** {  
 Toast.*makeText*(getApplicationContext(),**" network error to create acc"**,Toast.***LENGTH\_SHORT***).show();  
 **lodingBar**.dismiss();  
 }  
 }  
 });  
  
 }  
 **else** {  
 Toast.*makeText*(getApplicationContext(),**"user already exists with this number "**,Toast.***LENGTH\_SHORT***).show();  
 **lodingBar**.dismiss();  
 }  
 }  
  
 @Override  
 **public void** onCancelled(DatabaseError databaseError) {  
 Toast.*makeText*(getApplicationContext(),**" "**+databaseError,Toast.***LENGTH\_SHORT***).show();  
 **lodingBar**.dismiss();  
 }  
 });  
 }  
}

3.MainActivity.java

**package** com.allandroidprojects.buildsmart.startup;  
  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.os.Handler;  
**import** android.support.annotation.NonNull;  
**import** android.support.design.widget.NavigationView;  
**import** android.support.design.widget.TabLayout;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentManager;  
**import** android.support.v4.app.FragmentPagerAdapter;  
**import** android.support.v4.view.GravityCompat;  
**import** android.support.v4.view.ViewPager;  
**import** android.support.v4.widget.DrawerLayout;  
**import** android.support.v7.app.ActionBarDrawerToggle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.support.v7.widget.Toolbar;  
**import** android.view.Menu;  
**import** android.view.MenuItem;  
**import** android.widget.TextView;  
  
**import** com.allandroidprojects.buildsmart.R;  
**import** com.allandroidprojects.buildsmart.fragments.ImageListFragment;  
**import** com.allandroidprojects.buildsmart.imgmodel.ImageModel;  
**import** com.allandroidprojects.buildsmart.imgmodel.SlidingImage\_Adapter;  
**import** com.allandroidprojects.buildsmart.miscellaneous.EmptyActivity;  
**import** com.allandroidprojects.buildsmart.notification.NotificationCountSetClass;  
**import** com.allandroidprojects.buildsmart.options.CartListActivity;  
**import** com.allandroidprojects.buildsmart.options.SearchResultActivity;  
**import** com.allandroidprojects.buildsmart.options.WishlistActivity;  
**import** com.google.firebase.database.DataSnapshot;  
**import** com.google.firebase.database.DatabaseError;  
**import** com.google.firebase.database.DatabaseReference;  
**import** com.google.firebase.database.FirebaseDatabase;  
**import** com.google.firebase.database.ValueEventListener;  
**import** com.viewpagerindicator.CirclePageIndicator;  
  
**import** java.util.ArrayList;  
**import** java.util.List;  
**import** java.util.Timer;  
**import** java.util.TimerTask;  
  
**public class** MainActivity **extends** AppCompatActivity  
 **implements** NavigationView.OnNavigationItemSelectedListener {  
  
 **public static int** *notificationCountCart* = 0;  
 **static** ViewPager *viewPager*;  
 **static** TabLayout *tabLayout*;  
 String **name**;  
*// TextView tt=findViewById(R.id.tdraw);* **private static** ViewPager *mPager*;  
 **private static int** *currentPage* = 0;  
 **private static int** *NUM\_PAGES* = 0;  
 **private** ArrayList<ImageModel> **imageModelArrayList**;  
 String **s**;  
 **final** FirebaseDatabase **database** = FirebaseDatabase.*getInstance*();  
 DatabaseReference **ref**;  
  
  
 **private int**[] **myImageList** = **new int**[]{R.drawable.***ab***, R.drawable.***ac***,  
 R.drawable.***ad***,R.drawable.***ae*** ,R.drawable.***ab***,R.drawable.***ac***};  
  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 Intent i=getIntent();  
 **s**=i.getStringExtra(**"phnno"**);  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_mainf***);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.***toolbar***);  
 setSupportActionBar(toolbar);  
  
  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 ActionBarDrawerToggle toggle = **new** ActionBarDrawerToggle(  
 **this**, drawer, toolbar, R.string.***navigation\_drawer\_open***, R.string.***navigation\_drawer\_close***);  
 drawer.setDrawerListener(toggle);  
 toggle.syncState();  
  
 NavigationView navigationView = (NavigationView) findViewById(R.id.***nav\_view***);  
 navigationView.setNavigationItemSelectedListener(**this**);  
  
  
  
 **imageModelArrayList** = **new** ArrayList<>();  
 **imageModelArrayList** = populateList();  
  
 init();  
  
  
  
 *viewPager* = (ViewPager) findViewById(R.id.***viewpager***);  
 *tabLayout* = (TabLayout) findViewById(R.id.***tabs***);  
  
 **if** (*viewPager* != **null**) {  
 setupViewPager(*viewPager*);  
 *tabLayout*.setupWithViewPager(*viewPager*);  
 }  
  
  
  
 **private** ArrayList<ImageModel> populateList(){  
  
 ArrayList<ImageModel> list = **new** ArrayList<>();  
  
 **for**(**int** i = 0; i < 6; i++){  
 ImageModel imageModel = **new** ImageModel();  
 imageModel.setImage\_drawable(**myImageList**[i]);  
 list.add(imageModel);  
 }  
  
 **return** list;  
 }  
  
 **private void** init() {  
  
 *mPager* = (ViewPager) findViewById(R.id.***pager***);  
 *mPager*.setAdapter(**new** SlidingImage\_Adapter(MainActivity.**this**,**imageModelArrayList**));  
  
 CirclePageIndicator indicator = (CirclePageIndicator)  
 findViewById(R.id.***indicator***);  
  
 indicator.setViewPager(*mPager*);  
  
 **final float** density = getResources().getDisplayMetrics().**density**;  
  
*//Set circle indicator radius* indicator.setRadius(5 \* density);  
  
 *NUM\_PAGES* =**imageModelArrayList**.size();  
  
 *// Auto start of viewpager* **final** Handler handler = **new** Handler();  
 **final** Runnable Update = **new** Runnable() {  
 **public void** run() {  
 **if** (*currentPage* == *NUM\_PAGES*) {  
 *currentPage* = 0;  
 }  
 *mPager*.setCurrentItem(*currentPage*++, **true**);  
 }  
 };  
 Timer swipeTimer = **new** Timer();  
 swipeTimer.schedule(**new** TimerTask() {  
 @Override  
 **public void** run() {  
 handler.post(Update);  
 }  
 }, 1000, 1000);  
  
 *// Pager listener over indicator* indicator.setOnPageChangeListener(**new** ViewPager.OnPageChangeListener() {  
  
 @Override  
 **public void** onPageSelected(**int** position) {  
 *currentPage* = position;  
  
 }  
  
 @Override  
 **public void** onPageScrolled(**int** pos, **float** arg1, **int** arg2) {  
  
 }  
  
 @Override  
 **public void** onPageScrollStateChanged(**int** pos) {  
  
 }  
 });  
  
 }  
  
  
  
  
  
  
 @Override  
 **protected void** onResume() {  
 **super**.onResume();  
 invalidateOptionsMenu();  
 }  
  
 @Override  
 **public void** onBackPressed() {  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 **if** (drawer.isDrawerOpen(GravityCompat.***START***)) {  
 drawer.closeDrawer(GravityCompat.***START***);  
 } **else** {  
 **super**.onBackPressed();  
 }  
 }  
  
 @Override  
 **public boolean** onCreateOptionsMenu(Menu menu) {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.***main***, menu);  
 **return true**;  
 }  
  
 @Override  
 **public boolean** onPrepareOptionsMenu(Menu menu) {  
 *// Get the notifications MenuItem and  
 // its LayerDrawable (layer-list)* MenuItem item = menu.findItem(R.id.***action\_cart***);  
 NotificationCountSetClass.*setAddToCart*(MainActivity.**this**, item,*notificationCountCart*);  
 *// force the ActionBar to relayout its MenuItems.  
 // onCreateOptionsMenu(Menu) will be called again.* invalidateOptionsMenu();  
 **return super**.onPrepareOptionsMenu(menu);  
 }  
  
 @Override  
 **public boolean** onOptionsItemSelected(MenuItem item) {  
 *// Handle action bar item clicks here. The action bar will  
 // automatically handle clicks on the Home/Up button, so long  
 // as you specify a parent activity in AndroidManifest.xml.* **int** id = item.getItemId();  
  
 *//noinspection SimplifiableIfStatement* **if** (id == R.id.***action\_search***) {  
 startActivity(**new** Intent(MainActivity.**this**, SearchResultActivity.**class**));  
 **return true**;  
 }**else if** (id == R.id.***action\_cart***) {  
  
**return true**;  
 }**else** {  
 startActivity(**new** Intent(MainActivity.**this**, EmptyActivity.**class**));  
  
 }  
 **return super**.onOptionsItemSelected(item);  
 }  
  
 **private void** setupViewPager(ViewPager viewPager) {  
 Adapter adapter = **new** Adapter(getSupportFragmentManager());  
 ImageListFragment fragment = **new** ImageListFragment();  
 Bundle bundle = **new** Bundle();  
 bundle.putInt(**"type"**, 1);  
 fragment.setArguments(bundle);  
 adapter.addFragment(fragment, getString(R.string.***item\_1***));  
 fragment = **new** ImageListFragment();  
 bundle = **new** Bundle();  
 bundle.putInt(**"type"**, 2);  
 fragment.setArguments(bundle);  
 adapter.addFragment(fragment, getString(R.string.***item\_2***));  
 fragment = **new** ImageListFragment();  
 bundle = **new** Bundle();  
 bundle.putInt(**"type"**, 3);  
 fragment.setArguments(bundle);  
 adapter.addFragment(fragment, getString(R.string.***item\_3***));  
 fragment = **new** ImageListFragment();  
 bundle = **new** Bundle();  
 bundle.putInt(**"type"**, 4);  
 fragment.setArguments(bundle);  
 adapter.addFragment(fragment, getString(R.string.***item\_4***));  
 fragment = **new** ImageListFragment();  
 bundle = **new** Bundle();  
 bundle.putInt(**"type"**, 5);  
 fragment.setArguments(bundle);  
 adapter.addFragment(fragment, getString(R.string.***item\_5***));  
 fragment = **new** ImageListFragment();  
 bundle = **new** Bundle();  
 bundle.putInt(**"type"**, 6);  
 fragment.setArguments(bundle);  
 adapter.addFragment(fragment, getString(R.string.***item\_6***));  
 viewPager.setAdapter(adapter);  
 }  
  
 @SuppressWarnings(**"StatementWithEmptyBody"**)  
 @Override  
 **public boolean** onNavigationItemSelected(MenuItem item) {  
 *// Handle navigation view item clicks here.* **int** id = item.getItemId();  
  
 **if** (id == R.id.***nav\_item1***) {  
 *viewPager*.setCurrentItem(0);  
 } **else if** (id == R.id.***nav\_item2***) {  
 *viewPager*.setCurrentItem(1);  
 } **else if** (id == R.id.***nav\_item3***) {  
 *viewPager*.setCurrentItem(2);  
 } **else if** (id == R.id.***nav\_item4***) {  
 *viewPager*.setCurrentItem(3);  
 } **else if** (id == R.id.***nav\_item5***) {  
 *viewPager*.setCurrentItem(4);  
 }**else if** (id == R.id.***nav\_item6***) {  
 *viewPager*.setCurrentItem(5);  
 }**else if** (id == R.id.***my\_wishlist***) {  
 startActivity(**new** Intent(MainActivity.**this**, WishlistActivity.**class**));  
 }**else if** (id == R.id.***my\_cart***) {  
 startActivity(**new** Intent(MainActivity.**this**, CartListActivity.**class**));  
 }**else** {  
 startActivity(**new** Intent(MainActivity.**this**, EmptyActivity.**class**));  
 }  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 drawer.closeDrawer(GravityCompat.***START***);  
 **return true**;  
 }  
  
 **static class** Adapter **extends** FragmentPagerAdapter {  
 **private final** List<Fragment> **mFragments** = **new** ArrayList<>();  
 **private final** List<String> **mFragmentTitles** = **new** ArrayList<>();  
  
 **public** Adapter(FragmentManager fm) {  
 **super**(fm);  
 }  
  
 **public void** addFragment(Fragment fragment, String title) {  
 **mFragments**.add(fragment);  
 **mFragmentTitles**.add(title);  
 }  
  
 @Override  
 **public** Fragment getItem(**int** position) {  
 **return mFragments**.get(position);  
 }  
  
 @Override  
 **public int** getCount() {  
 **return mFragments**.size();  
 }  
  
 @Override  
 **public** CharSequence getPageTitle(**int** position) {  
 **return mFragmentTitles**.get(position);  
 }  
 }  
}

4.Utility

4.a ImageUrlUtils

**package** com.allandroidprojects.buildsmart.utility;  
  
**import** com.allandroidprojects.buildsmart.DisplayHome.HomeActivity;  
  
**import** java.util.ArrayList;  
  
*/\*\*  
 \* Created by 06peng on 2015/6/24.  
 \*/***public class** ImageUrlUtils {  
 **static** ArrayList<String> *wishlistImageUri* = **new** ArrayList<>();  
 **static** ArrayList<String> *cartListImageUri* = **new** ArrayList<>();  
 **static** ArrayList<String> *cartListpName* = **new** ArrayList<>();  
 **static** ArrayList<String> *cartListpPrice* = **new** ArrayList<>();  
  
 **public static** String[] getMoreDesc()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**moreDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**moreDetails**.get(i).getDescription();}  
 **return** names;  
  
 }  
 **public static** String[] getMobilesDesc()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**mobilesDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**mobilesDetails**.get(i).getDescription();}  
 **return** names;  
  
 }  
 **public static** String[] getOffersDesc()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**offerDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**offerDetails**.get(i).getDescription();}  
 **return** names;  
  
 }  
 **public static** String[] getLifestyleDesc()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**lifestyleDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**lifestyleDetails**.get(i).getDescription();}  
 **return** names;  
  
 }  
 **public static** String[] getBooksDesc()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**booksDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**booksDetails**.get(i).getDescription();}  
 **return** names;  
  
 }  
 **public static** String[] getLaptopsDesc()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**laptopsDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**laptopsDetails**.get(i).getDescription();}  
 **return** names;  
  
 }  
  
  
 **public static** String[] getMoreName()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**moreDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**moreDetails**.get(i).getName();}  
 **return** names;  
  
 }  
 **public static** String[] getMorePrice()  
 {  
 **int** size= HomeActivity.*getInstanc*().**booksDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**booksDetails**.get(i).getPrice();}  
 **return** names;  
  
  
 }  
 **public static** String[] getBooksPrice()  
 {  
 **int** size= HomeActivity.*getInstanc*().**booksDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**booksDetails**.get(i).getPrice();}  
 **return** names;  
  
  
 }  
  
  
 **public static** String[] getBooksName()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**booksDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**booksDetails**.get(i).getName();}  
 **return** names;  
  
 }  
  
  
 **public static** String[] getMobilesName()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**mobilesDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**mobilesDetails**.get(i).getName();}  
 **return** names;  
  
 }  
 **public static** String[] getMobilesPrice()  
 {  
 **int** size= HomeActivity.*getInstanc*().**mobilesDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**mobilesDetails**.get(i).getPrice();}  
 **return** names;  
  
  
 }  
  
 **public static** String[] getLifestyleName()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**lifestyleDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**lifestyleDetails**.get(i).getName();}  
 **return** names;  
  
 }  
 **public static** String[] getLifestylePrice()  
 {  
 **int** size= HomeActivity.*getInstanc*().**lifestyleDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**lifestyleDetails**.get(i).getPrice();}  
 **return** names;  
  
  
 }  
  
 **public static** String[] getLaptopsName()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**laptopsDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**laptopsDetails**.get(i).getName();}  
 **return** names;  
  
 }  
 **public static** String[] getLaptopsPrice()  
 {  
 **int** size= HomeActivity.*getInstanc*().**laptopsDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**laptopsDetails**.get(i).getPrice();}  
 **return** names;  
  
  
 }  
  
 **public static** String[] getOffersName()  
 {  
  
 **int** size= HomeActivity.*getInstanc*().**offerDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**offerDetails**.get(i).getName();}  
 **return** names;  
  
 }  
 **public static** String[] getOffersPrice()  
 {  
 **int** size= HomeActivity.*getInstanc*().**offerDetails**.size();  
 String []names = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 names[i] =HomeActivity.*getInstanc*().**offerDetails**.get(i).getPrice();}  
 **return** names;  
  
  
 }  
  
  
  
  
  
 **public static** String[] getImageUrls() {  
 **int** size= HomeActivity.*getInstanc*().**moreDetails**.size();  
 String []urls = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 urls[i] =HomeActivity.*getInstanc*().**moreDetails**.get(i).getDownloadurl();}  
  
**return** urls;  
 }  
  
 **public static** String[] getOffersUrls() {  
  
 **int** size= HomeActivity.*getInstanc*().**offerDetails**.size();  
 String []urls = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 urls[i] =HomeActivity.*getInstanc*().**offerDetails**.get(i).getDownloadurl();}  
**return** urls;  
 }  
  
 **public static** String[] getMobilesUrls() {  
 **int** size= HomeActivity.*getInstanc*().**mobilesDetails**.size();  
 String []urls = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 urls[i] =HomeActivity.*getInstanc*().**mobilesDetails**.get(i).getDownloadurl();}  
**return** urls;  
 }  
  
 **public static** String[] getLaptopsUrls() {  
 **int** size= HomeActivity.*getInstanc*().**laptopsDetails**.size();  
 String []urls = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 urls[i] =HomeActivity.*getInstanc*().**laptopsDetails**.get(i).getDownloadurl();}  
**return** urls;  
 }  
  
 **public static** String[] getLifeStyleUrls() {  
 **int** size= HomeActivity.*getInstanc*().**lifestyleDetails**.size();  
 String []urls = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 urls[i] =HomeActivity.*getInstanc*().**lifestyleDetails**.get(i).getDownloadurl();}  
**return** urls;  
 }  
  
 **public static** String[] getBooksUrls() {  
 **int** size= HomeActivity.*getInstanc*().**booksDetails**.size();  
 String []urls = **new** String [size];  
 **for**(**int** i=0;i<size;i++){  
 urls[i] =HomeActivity.*getInstanc*().**booksDetails**.get(i).getDownloadurl();}  
**return** urls;  
 }  
  
 *// Methods for Wishlist* **public void** addWishlistImageUri(String wishlistImageUri) {  
 **this**.*wishlistImageUri*.add(0,wishlistImageUri);  
 }  
  
 **public void** removeWishlistImageUri(**int** position) {  
 **this**.*wishlistImageUri*.remove(position);  
 }  
  
 **public** ArrayList<String> getWishlistImageUri(){ **return this**.*wishlistImageUri*; }  
  
 *// Methods for Cart* **public int** addCartListImageUri(String wishlistImageUri,String name,String price,**int** total) {  
 **this**.*cartListImageUri*.add(0,wishlistImageUri);  
 **this**.*cartListpName*.add(0,name);  
 **this**.*cartListpPrice*.add(0,price);  
 total=total+Integer.*parseInt*(price);  
 **return** total;  
 }  
  
  
 **public int** removeCartListImageUri(**int** position,**int** total) {  
 **this**.*cartListImageUri*.remove(position);  
 total=total-Integer.*parseInt*(*cartListpPrice*.get(position));  
 **return** total;  
 }  
  
 **public** ArrayList<String> getCartListImageUri(){ **return this**.*cartListImageUri*; }  
  
 **public** ArrayList<String> getppriceCartList(){ **return this**.*cartListpName*; }  
  
 **public** ArrayList<String> getpnameCartList(){ **return this**.*cartListpPrice*; }  
  
  
}

5.Product

ItemdetailsActivity.java

**package** com.allandroidprojects.buildsmart.product;  
  
**import** android.content.Intent;  
**import** android.net.Uri;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
**import** com.allandroidprojects.buildsmart.R;  
**import** com.allandroidprojects.buildsmart.fragments.ImageListFragment;  
**import** com.allandroidprojects.buildsmart.fragments.ViewPagerActivity;  
**import** com.allandroidprojects.buildsmart.notification.NotificationCountSetClass;  
**import** com.allandroidprojects.buildsmart.options.CartListActivity;  
**import** com.allandroidprojects.buildsmart.startup.MainActivity;  
**import** com.allandroidprojects.buildsmart.utility.ImageUrlUtils;  
**import** com.facebook.drawee.view.SimpleDraweeView;  
  
**public class** ItemDetailsActivity **extends** AppCompatActivity {  
 **int imagePosition**;  
 **public static int** *total*=0;  
 String **stringImageUri**;  
 String **stringImageName**;  
 String **stringImagePrice**;  
 TextView **pname**,**pprice**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_item\_details***);  
 SimpleDraweeView mImageView = (SimpleDraweeView)findViewById(R.id.***image1***);  
 TextView textViewAddToCart = (TextView)findViewById(R.id.***text\_action\_bottom1***);  
 TextView textViewBuyNow = (TextView)findViewById(R.id.***text\_action\_bottom2***);  
  
  
 **pname**=findViewById(R.id.***productName***);  
 **pprice**=findViewById(R.id.***productPrice***);  
  
  
  
  
 *//Getting image uri from previous screen* **if** (getIntent() != **null**) {  
 **stringImageUri** = getIntent().getStringExtra(ImageListFragment.***STRING\_IMAGE\_URI***);  
 **imagePosition** = getIntent().getIntExtra(ImageListFragment.***STRING\_IMAGE\_URI***,0);  
 **stringImageName**=getIntent().getStringExtra(ImageListFragment.***STRING\_IMAGE\_NAME***);  
 **stringImagePrice**=getIntent().getStringExtra(ImageListFragment.***STRING\_IMAGE\_PRICE***);  
 }  
  
 **pname**.setText(**stringImageName**);  
 **pprice**.setText(**stringImagePrice**);  
 Uri uri = Uri.*parse*(**stringImageUri**);  
 mImageView.setImageURI(uri);  
 mImageView.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent intent = **new** Intent(ItemDetailsActivity.**this**, ViewPagerActivity.**class**);  
 intent.putExtra(**"position"**, **imagePosition**);  
 startActivity(intent);  
  
 }  
 });  
  
 textViewAddToCart.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 ImageUrlUtils imageUrlUtils = **new** ImageUrlUtils();  
 *total*=imageUrlUtils.addCartListImageUri(**stringImageUri**,**stringImageName**,**stringImagePrice**,*total*);  
  
 Toast.*makeText*(ItemDetailsActivity.**this**,**"Item added to cart."**,Toast.***LENGTH\_SHORT***).show();  
 MainActivity.*notificationCountCart*++;  
 NotificationCountSetClass.*setNotifyCount*(MainActivity.*notificationCountCart*);  
 }  
 });  
  
 textViewBuyNow.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 ImageUrlUtils imageUrlUtils = **new** ImageUrlUtils();  
 *total*=imageUrlUtils.addCartListImageUri(**stringImageUri**,**stringImageName**,**stringImagePrice**,*total*);  
 MainActivity.*notificationCountCart*++;  
 NotificationCountSetClass.*setNotifyCount*(MainActivity.*notificationCountCart*);  
 startActivity(**new** Intent(ItemDetailsActivity.**this**, CartListActivity.**class**));  
  
 }  
 });  
  
 }  
}

6.Category

6.a AddCategory

**package** com.allandroidprojects.buildsmart.Category;  
  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.view.View;  
**import** android.widget.ImageView;  
  
**import** com.allandroidprojects.buildsmart.R;  
  
  
**public class** AddCategory **extends** AppCompatActivity {  
**private** ImageView **Offers**,**Laptops**,**Mobiles**,**More**;  
**private** ImageView **Purses**,**Lifestyle**,**Shoes**,**Sports**;  
**private** ImageView **Glasses**,**Books**,**Hats**,**Sweaters**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_add\_category***);  
  
 **Offers**=(ImageView)findViewById(R.id.***headphones***);  
 **Laptops**=findViewById(R.id.***laptops***);  
 **Mobiles**=findViewById(R.id.***mobiles***);  
 **More**=findViewById(R.id.***more***);  
  
 **Purses**=findViewById(R.id.***purses***);  
 **Lifestyle**=findViewById(R.id.***tshirts***);  
 **Shoes**=findViewById(R.id.***shoes***);  
 **Sports**=findViewById(R.id.***sports***);  
  
 **Glasses**=findViewById(R.id.***glasses***);  
 **Books**=findViewById(R.id.***books***);  
 **Hats**=findViewById(R.id.***hats***);  
 **Sweaters**=findViewById(R.id.***sweaters***);  
  
 **Offers**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"Offers"**);  
 startActivity(i);  
 }  
 });  
 **Laptops**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"Laptop"**);  
 startActivity(i);  
 }  
 });  
 **Mobiles**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"Mobile"**);  
 startActivity(i);  
 }  
 });  
 **More**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"More"**);  
 startActivity(i);  
 }  
 });  
 **Purses**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"Purse"**);  
 startActivity(i);  
 }  
 });  
  
  
 **Lifestyle**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"Lifestyle"**);  
 startActivity(i);  
 }  
 });  
 **Shoes**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"Shoe"**);  
 startActivity(i);  
 }  
 });  
 **Sports**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"Sport"**);  
 startActivity(i);  
 }  
 });  
 **Glasses**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"Glasses"**);  
 startActivity(i);  
 }  
 });  
  
  
 **Hats**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"Hat"**);  
 startActivity(i);  
 }  
 });  
  
  
 **Books**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"Book"**);  
 startActivity(i);  
 }  
 });  
 **Sweaters**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent i=**new** Intent(AddCategory.**this**,UploadCategoryItem.**class**);  
 i.putExtra(**"name"**,**"Sweater"**);  
 startActivity(i);  
 }  
 });  
  
 }  
}

6.b UploadCategoryItem

**package** com.allandroidprojects.buildsmart.Category;  
  
**import** android.app.ProgressDialog;  
**import** android.content.Intent;  
**import** android.net.Uri;  
**import** android.os.Bundle;  
**import** android.support.annotation.NonNull;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.text.TextUtils;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.ImageView;  
**import** android.widget.Toast;  
  
**import** com.allandroidprojects.buildsmart.R;  
**import** com.google.android.gms.tasks.OnCompleteListener;  
**import** com.google.android.gms.tasks.OnFailureListener;  
**import** com.google.android.gms.tasks.OnSuccessListener;  
**import** com.google.android.gms.tasks.Task;  
**import** com.google.firebase.database.DatabaseReference;  
**import** com.google.firebase.database.FirebaseDatabase;  
**import** com.google.firebase.storage.FirebaseStorage;  
**import** com.google.firebase.storage.StorageReference;  
**import** com.google.firebase.storage.UploadTask;  
  
**import** java.text.SimpleDateFormat;  
**import** java.util.Calendar;  
**import** java.util.HashMap;  
  
**public class** UploadCategoryItem **extends** AppCompatActivity {  
**private** ImageView **selectImage**;  
**private** EditText **pname**,**pprice**,**pdescription**;  
**private static final int *requestCode***=1;  
**private** Uri **imageUri**;  
**private** Button **upload**;  
**private** String **productName**,**productPrice**,**productDescription**;  
**private int flag**=0;  
**private** StorageReference **storageRef**;  
**private** DatabaseReference **myref**;  
  
**private** String **saveCurrentTime**,**downloadUrl**,**category**;  
**private** String **saveCurrentDate**,**productRandomTitle**;  
**private** ProgressDialog **progressDialog**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_upload\_category\_item***);  
 **selectImage**=findViewById(R.id.***select\_image***);  
 **category**=getIntent().getStringExtra(**"name"**);  
 **upload**=findViewById(R.id.***upload***);  
 **progressDialog**=**new** ProgressDialog(**this**);  
  
 **pname**=findViewById(R.id.***productName***);  
 **pprice**=findViewById(R.id.***productPrice***);  
 **pdescription**=findViewById(R.id.***ProductDescription***);  
 **selectImage**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 selectImage();  
  
 }  
 });  
 **upload**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 validateInputs();  
 }  
 });  
 }  
  
 **private void** validateInputs() {  
 **productName**=**pname**.getText().toString();  
 **productPrice**=**pprice**.getText().toString();  
 **productDescription**=**pdescription**.getText().toString();  
  
 **if**(TextUtils.*isEmpty*(**productName**))  
 {  
 **flag**=1;  
 }  
 **else if**(**imageUri**==**null**)  
 {  
 **flag**=1;  
 }  
 **else if**(TextUtils.*isEmpty*(**productPrice**))  
 {  
 **flag**=1;  
 }  
 **else if**(TextUtils.*isEmpty*(**productDescription**))  
 {  
 **flag**=1;  
 }  
  
 **else** {  
 **flag**=0;  
 }  
 **if**(**flag**==1)  
 {  
 Toast.*makeText*(getApplicationContext(),**"Fill all the fields "**,Toast.***LENGTH\_SHORT***).show();}  
 **else** {  
 storeProductInformation();  
 }  
 }  
  
 **private void** storeProductInformation() {  
 **storageRef** = FirebaseStorage.*getInstance*().getReference().child(**"ProductImage"**);  
  
 Calendar calendar = Calendar.*getInstance*();  
 SimpleDateFormat currentDate=**new** SimpleDateFormat(**"MMM dd, yyyy"**);  
 **saveCurrentDate** =currentDate.format(calendar.getTime());  
  
 SimpleDateFormat currentTime=**new** SimpleDateFormat(**"HH:mm:ss a"**);  
 **saveCurrentTime**=currentTime.format(calendar.getTime());  
  
 **productRandomTitle**=**saveCurrentDate**+**saveCurrentTime**;  
  
 **final** StorageReference filePath= **storageRef**.child(**imageUri**.getLastPathSegment()+**productRandomTitle**+**".jpg"**);  
 **final** UploadTask uploadTask=filePath.putFile(**imageUri**);  
 **progressDialog**.setTitle(**"Uploading.."**);  
 **progressDialog**.setMessage(**"uploading your selected image wait.."**);  
 **progressDialog**.show();  
 uploadTask.addOnFailureListener(**new** OnFailureListener() {  
 @Override  
 **public void** onFailure(@NonNull Exception e) {  
 Toast.*makeText*(getApplicationContext(),**"error : "**+e,Toast.***LENGTH\_LONG***).show();  
 **progressDialog**.dismiss();  
 }  
 }).addOnSuccessListener(**new** OnSuccessListener<UploadTask.TaskSnapshot>() {  
 @Override  
 **public void** onSuccess(UploadTask.TaskSnapshot taskSnapshot) {  
  
 filePath.getDownloadUrl().addOnSuccessListener(**new** OnSuccessListener<Uri>() {  
 @Override  
 **public void** onSuccess(Uri uri) {  
 Toast.*makeText*(getApplicationContext(),**"successfully inserted your image into database "**,Toast.***LENGTH\_LONG***).show();  
 **downloadUrl**=uri.toString();  
 productDetailsIntoDatabase();  
 }  
 });  
  
  
*// downloadUrl = taskSnapshot.getDownloadUrl().toString();* }  
 });  
 }  
  
 **private void** productDetailsIntoDatabase() {  
 **myref** = FirebaseDatabase.*getInstance*().getReference(**"products"**);  
  
 HashMap<String,Object> productDetails=**new** HashMap<>();  
 productDetails.put(**"pid"**,**productRandomTitle**);  
 productDetails.put(**"date"**,**saveCurrentDate**);  
 productDetails.put(**"time"**,**saveCurrentTime**);  
 productDetails.put(**"name"**,**productName**);  
 productDetails.put(**"Description"**,**productDescription**);  
 productDetails.put(**"price"**,**productPrice**);  
 productDetails.put(**"downloadurl"**,**downloadUrl**);  
 productDetails.put(**"category"**,**category**);  
 **myref**.child(**category**).child(**productRandomTitle**).updateChildren(productDetails).addOnCompleteListener(**new** OnCompleteListener<Void>() {  
 @Override  
 **public void** onComplete(@NonNull Task<Void> task) {  
 **if**(task.isSuccessful())  
 {  
 Toast.*makeText*(getApplicationContext(),**"successfully inserted product details into database "**,Toast.***LENGTH\_LONG***).show();  
 Intent i=**new** Intent(getApplicationContext(),AddCategory.**class**);  
 startActivity(i);  
 display\_home();  
 }  
 **else** { String e=task.getException().toString();  
 Toast.*makeText*(getApplicationContext(),**"error: "**+e,Toast.***LENGTH\_LONG***).show();  
 }  
 }  
 });  
 **progressDialog**.dismiss();  
  
 }  
  
 **private void** display\_home() {  
  
 }  
  
 **private void** selectImage()  
 {  
 Intent i=**new** Intent(Intent.***ACTION\_GET\_CONTENT***);  
 i.setType(**"image/\*"**);  
 startActivityForResult(i,***requestCode***);  
 }  
  
 @Override  
 **protected void** onActivityResult(**int** requestCode, **int** resultCode, Intent data) {  
 **super**.onActivityResult(requestCode, resultCode, data);  
 **if**(requestCode==**this**.***requestCode*** && resultCode==***RESULT\_OK***)  
 {  
 **imageUri**=data.getData();  
 **selectImage**.setImageURI(**imageUri**);  
  
 }  
 }  
}

7.ImageListFragments.java

**package** com.allandroidprojects.buildsmart.fragments;  
  
**import** android.content.Intent;  
**import** android.net.Uri;  
**import** android.os.Bundle;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v7.widget.RecyclerView;  
**import** android.support.v7.widget.StaggeredGridLayoutManager;  
**import** android.util.EventLogTags;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.ImageView;  
**import** android.widget.LinearLayout;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** com.allandroidprojects.buildsmart.R;  
**import** com.allandroidprojects.buildsmart.product.ItemDetailsActivity;  
**import** com.allandroidprojects.buildsmart.startup.MainActivity;  
**import** com.allandroidprojects.buildsmart.utility.ImageUrlUtils;  
**import** com.facebook.drawee.view.SimpleDraweeView;  
  
  
**public class** ImageListFragment **extends** Fragment {  
  
 **public static final** String ***STRING\_IMAGE\_URI*** = **"ImageUri"**;  
 **public static final** String ***STRING\_IMAGE\_POSITION*** = **"ImagePosition"**;  
 **private static** MainActivity *mActivity*;  
  
 **public static final** String ***STRING\_IMAGE\_NAME*** = **"ImageName"**;  
 **public static final** String ***STRING\_IMAGE\_DESC*** = **"ImageDesc"**;  
 **public static final** String ***STRING\_IMAGE\_PRICE*** = **"ImagePrice"**;  
  
 @Override  
 **public void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 *mActivity* = (MainActivity) getActivity();  
 }  
  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {  
 RecyclerView rv = (RecyclerView) inflater.inflate(R.layout.***layout\_recylerview\_list***, container, **false**);  
 setupRecyclerView(rv);  
 **return** rv;  
 }  
  
 **private void** setupRecyclerView(RecyclerView recyclerView) {  
String[] items=**null**;  
 String pname[]=**null**;  
 String pprice[]=**null**;  
 String pdesc[]=**null**;  
 **if** (ImageListFragment.**this**.getArguments().getInt(**"type"**) == 1){  
 items =ImageUrlUtils.*getOffersUrls*();  
 pname=ImageUrlUtils.*getOffersName*();  
 pprice=ImageUrlUtils.*getOffersPrice*();  
 pdesc=ImageUrlUtils.*getOffersDesc*();  
  
 }**else if** (ImageListFragment.**this**.getArguments().getInt(**"type"**) == 2){  
 items =ImageUrlUtils.*getLaptopsUrls*();  
 pname=ImageUrlUtils.*getLaptopsName*();  
 pprice=ImageUrlUtils.*getLaptopsPrice*();  
 pdesc=ImageUrlUtils.*getLaptopsDesc*();  
 }**else if** (ImageListFragment.**this**.getArguments().getInt(**"type"**) == 3){  
 items =ImageUrlUtils.*getLifeStyleUrls*();  
 pname=ImageUrlUtils.*getLifestyleName*();  
 pprice=ImageUrlUtils.*getLifestylePrice*();  
 pdesc=ImageUrlUtils.*getLifestyleDesc*();  
 }**else if** (ImageListFragment.**this**.getArguments().getInt(**"type"**) == 4){  
 items =ImageUrlUtils.*getMobilesUrls*();  
 pname=ImageUrlUtils.*getMobilesName*();  
 pprice=ImageUrlUtils.*getMobilesPrice*();  
 pdesc=ImageUrlUtils.*getMobilesDesc*();  
 }**else if** (ImageListFragment.**this**.getArguments().getInt(**"type"**) == 5){  
 items =ImageUrlUtils.*getBooksUrls*();  
 pname=ImageUrlUtils.*getBooksName*();  
 pprice=ImageUrlUtils.*getBooksPrice*();  
 pdesc=ImageUrlUtils.*getBooksDesc*();  
 }**else** {  
 items = ImageUrlUtils.*getImageUrls*();  
 pname=ImageUrlUtils.*getMoreName*();  
 pprice=ImageUrlUtils.*getMorePrice*();  
 pdesc=ImageUrlUtils.*getMoreDesc*();  
 }  
 StaggeredGridLayoutManager layoutManager = **new** StaggeredGridLayoutManager(2, StaggeredGridLayoutManager.***VERTICAL***);  
 recyclerView.setLayoutManager(layoutManager);  
 recyclerView.setAdapter(**new** SimpleStringRecyclerViewAdapter(recyclerView, items,pname,pprice,pdesc));  
 }  
  
 **public static class** SimpleStringRecyclerViewAdapter  
 **extends** RecyclerView.Adapter<SimpleStringRecyclerViewAdapter.ViewHolder> {  
  
 **private** String[] **mValues**;  
 **private** String[] **proname**;  
 **private** String[] **pdesc**;  
 **private** String[] **proprice**;  
 **private** RecyclerView **mRecyclerView**;  
  
 **public static class** ViewHolder **extends** RecyclerView.ViewHolder {  
 **public final** View **mView**;  
 **public final** SimpleDraweeView **mImageView**;  
 **public final** LinearLayout **mLayoutItem**;  
 **public final** ImageView **mImageViewWishlist**;  
 **public final** TextView **tprice**;  
 **public final** TextView **tname**;  
 **public final** TextView **desc**;  
 **public** ViewHolder(View view) {  
 **super**(view);  
 **mView** = view;  
 **mImageView** = (SimpleDraweeView) view.findViewById(R.id.***image1***);  
 **mLayoutItem** = (LinearLayout) view.findViewById(R.id.***layout\_item***);  
 **mImageViewWishlist** = (ImageView) view.findViewById(R.id.***ic\_wishlist***);  
 **tprice**=(TextView)view.findViewById(R.id.***pprice***);  
 **tname**=(TextView)view.findViewById(R.id.***pname***);  
 **desc**=(TextView)view.findViewById(R.id.***description***);  
 }  
  
  
 }  
  
 **public** SimpleStringRecyclerViewAdapter(RecyclerView recyclerView, String[] items,String pname[],String pprice[],String desc[]) {  
 **mValues** = items;  
 **proprice**=pprice;  
 **proname**=pname;  
 **mRecyclerView** = recyclerView;  
 **pdesc**=desc;  
 }  
  
 @Override  
 **public** ViewHolder onCreateViewHolder(ViewGroup parent, **int** viewType) {  
 View view = LayoutInflater.*from*(parent.getContext()).inflate(R.layout.***list\_item***, parent, **false**);  
 **return new** ViewHolder(view);  
 }  
  
 @Override  
 **public void** onViewRecycled(ViewHolder holder) {  
 **if** (holder.**mImageView**.getController() != **null**) {  
 holder.**mImageView**.getController().onDetach();  
 }  
 **if** (holder.**mImageView**.getTopLevelDrawable() != **null**) {  
 holder.**mImageView**.getTopLevelDrawable().setCallback(**null**);  
*// ((BitmapDrawable) holder.mImageView.getTopLevelDrawable()).getBitmap().recycle();* }  
 }  
  
 @Override  
 **public void** onBindViewHolder(**final** ViewHolder holder, **final int** position) {  
 holder.**tprice**.setText(**proprice**[position]);  
 holder.**tname**.setText(**proname**[position]);  
 holder.**desc**.setText(**pdesc**[position]);  
 **final** Uri uri = Uri.*parse*(**mValues**[position]);  
 holder.**mImageView**.setImageURI(uri);  
 holder.**mLayoutItem**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent intent = **new** Intent(*mActivity*, ItemDetailsActivity.**class**);  
 intent.putExtra(***STRING\_IMAGE\_URI***, **mValues**[position]);  
 intent.putExtra(***STRING\_IMAGE\_POSITION***, position);  
  
  
 *//imgdetails* intent.putExtra(***STRING\_IMAGE\_NAME***, **proname**[position]);  
 intent.putExtra(***STRING\_IMAGE\_PRICE***, **proprice**[position]);  
 *mActivity*.startActivity(intent);  
  
 }  
 });  
  
 *//Set click action for wishlist* holder.**mImageViewWishlist**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 ImageUrlUtils imageUrlUtils = **new** ImageUrlUtils();  
 imageUrlUtils.addWishlistImageUri(**mValues**[position]);  
 holder.**mImageViewWishlist**.setImageResource(R.drawable.***ic\_favorite\_black\_18dp***);  
 notifyDataSetChanged();  
 Toast.*makeText*(*mActivity*,**"Item added to wishlist."**,Toast.***LENGTH\_SHORT***).show();  
  
 }  
 });  
  
 }  
  
 @Override  
 **public int** getItemCount() {  
 **return mValues**.**length**;  
 }  
 }  
}

8.Options

8.a CartListActivity

**package** com.allandroidprojects.buildsmart.options;  
  
**import** android.content.Context;  
**import** android.content.Intent;  
**import** android.net.Uri;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.support.v7.widget.LinearLayoutManager;  
**import** android.support.v7.widget.RecyclerView;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.Button;  
**import** android.widget.LinearLayout;  
**import** android.widget.TextView;  
  
**import** com.allandroidprojects.buildsmart.R;  
**import** com.allandroidprojects.buildsmart.payu\_de.MainActivity2;  
**import** com.allandroidprojects.buildsmart.payu\_de.StartPaymentActivity;  
**import** com.allandroidprojects.buildsmart.product.ItemDetailsActivity;  
**import** com.allandroidprojects.buildsmart.startup.MainActivity;  
**import** com.allandroidprojects.buildsmart.utility.ImageUrlUtils;  
**import** com.facebook.drawee.view.SimpleDraweeView;  
  
**import** java.util.ArrayList;  
  
**import static** com.allandroidprojects.buildsmart.fragments.ImageListFragment.***STRING\_IMAGE\_POSITION***;  
**import static** com.allandroidprojects.buildsmart.fragments.ImageListFragment.***STRING\_IMAGE\_URI***;  
**import static** com.allandroidprojects.buildsmart.product.ItemDetailsActivity.*total*;  
  
**public class** CartListActivity **extends** AppCompatActivity {  
 **private static** Context *mContext*;  
 **private static** TextView *disptotal*;  
 String **phone**=**"9756423780"**;  
 **int pay**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_cart\_list***);  
 *mContext* = CartListActivity.**this**;  
  
 *disptotal*=(TextView)findViewById(R.id.***text\_action\_bottom1***);  
 TextView textViewPayment = (TextView)findViewById(R.id.***text\_action\_bottom2***);  
 *disptotal*.setText(*total*+**""**);  
  
 ImageUrlUtils imageUrlUtils = **new** ImageUrlUtils();  
 ArrayList<String> cartlistImageUri =imageUrlUtils.getCartListImageUri();  
 ArrayList<String> cartlistName =imageUrlUtils.getpnameCartList();  
 ArrayList<String> cartlistPrice =imageUrlUtils.getppriceCartList();  
 *//Show cart layout based on items* setCartLayout();  
  
 RecyclerView recyclerView = (RecyclerView)findViewById(R.id.***recyclerview***);  
 RecyclerView.LayoutManager recylerViewLayoutManager = **new** LinearLayoutManager(*mContext*);  
  
 recyclerView.setLayoutManager(recylerViewLayoutManager);  
 recyclerView.setAdapter(**new** CartListActivity.SimpleStringRecyclerViewAdapter(recyclerView, cartlistImageUri,cartlistName,cartlistPrice));  
  
  
 textViewPayment.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
Intent in=**new** Intent(getApplicationContext(), StartPaymentActivity.**class**);  
  
 in.putExtra(**"amount"**,*total*+**""**);  
 in.putExtra(**"phone"**,**phone**);  
 startActivity(in);  
 }  
 });  
  
  
 }  
  
 **public static class** SimpleStringRecyclerViewAdapter  
 **extends** RecyclerView.Adapter<CartListActivity.SimpleStringRecyclerViewAdapter.ViewHolder> {  
  
 **private** ArrayList<String> **mCartlistImageUri**;  
 **private** ArrayList<String> **pnameCartList**;  
 **private** ArrayList<String> **ppriceCartList**;  
  
 **private** RecyclerView **mRecyclerView**;  
  
 **public static class** ViewHolder **extends** RecyclerView.ViewHolder {  
 **public final** View **mView**;  
 **public final** SimpleDraweeView **mImageView**;  
 **public final** TextView **pname**,**pprice**;  
 **public final** LinearLayout **mLayoutItem**, **mLayoutRemove** , **mLayoutEdit**;  
  
 **public** ViewHolder(View view) {  
 **super**(view);  
 **mView** = view;  
 **pname**=(TextView) view.findViewById(R.id.***itemname***);  
 **pprice**=(TextView) view.findViewById(R.id.***itemprice***);  
  
 **mImageView** = (SimpleDraweeView) view.findViewById(R.id.***image\_cartlist***);  
 **mLayoutItem** = (LinearLayout) view.findViewById(R.id.***layout\_item\_desc***);  
 **mLayoutRemove** = (LinearLayout) view.findViewById(R.id.***layout\_action1***);  
 **mLayoutEdit** = (LinearLayout) view.findViewById(R.id.***layout\_action2***);  
 }  
 }  
  
 **public** SimpleStringRecyclerViewAdapter(RecyclerView recyclerView, ArrayList<String> wishlistImageUri,ArrayList<String> a,ArrayList<String> b) {  
 **mCartlistImageUri** = wishlistImageUri;  
 **mRecyclerView** = recyclerView;  
 **pnameCartList**=a;  
 **ppriceCartList**=b;  
  
  
 }  
  
 @Override  
 **public** CartListActivity.SimpleStringRecyclerViewAdapter.ViewHolder onCreateViewHolder(ViewGroup parent, **int** viewType) {  
 View view = LayoutInflater.*from*(parent.getContext()).inflate(R.layout.***layout\_cartlist\_item***, parent, **false**);  
 **return new** CartListActivity.SimpleStringRecyclerViewAdapter.ViewHolder(view);  
 }  
  
 @Override  
 **public void** onViewRecycled(CartListActivity.SimpleStringRecyclerViewAdapter.ViewHolder holder) {  
 **if** (holder.**mImageView**.getController() != **null**) {  
 holder.**mImageView**.getController().onDetach();  
 }  
 **if** (holder.**mImageView**.getTopLevelDrawable() != **null**) {  
 holder.**mImageView**.getTopLevelDrawable().setCallback(**null**);  
*// ((BitmapDrawable) holder.mImageView.getTopLevelDrawable()).getBitmap().recycle();* }  
 }  
  
 @Override  
 **public void** onBindViewHolder(**final** CartListActivity.SimpleStringRecyclerViewAdapter.ViewHolder holder, **final int** position) {  
 **final** Uri uri = Uri.*parse*(**mCartlistImageUri**.get(position));  
 holder.**mImageView**.setImageURI(uri);  
 holder.**pname**.setText(**pnameCartList**.get(position));  
 holder.**pprice**.setText(**ppriceCartList**.get(position));  
 holder.**mLayoutItem**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent intent = **new** Intent(*mContext*, ItemDetailsActivity.**class**);  
 intent.putExtra(***STRING\_IMAGE\_URI***,**mCartlistImageUri**.get(position));  
 intent.putExtra(***STRING\_IMAGE\_POSITION***, position);  
 *mContext*.startActivity(intent);  
  
  
  
 }  
 });  
  
 *//Set click action* holder.**mLayoutRemove**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 ImageUrlUtils imageUrlUtils = **new** ImageUrlUtils();  
 *total*=imageUrlUtils.removeCartListImageUri(position,*total*);  
 *disptotal*.setText(*total*+**""**);  
 notifyDataSetChanged();  
 *//Decrease notification count* MainActivity.*notificationCountCart*--;  
  
 }  
 });  
  
 *//Set click action* holder.**mLayoutEdit**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 }  
 });  
 }  
  
 @Override  
 **public int** getItemCount() {  
 **return mCartlistImageUri**.size();  
 }  
 }  
  
 **protected void** setCartLayout(){  
 LinearLayout layoutCartItems = (LinearLayout) findViewById(R.id.***layout\_items***);  
 LinearLayout layoutCartPayments = (LinearLayout) findViewById(R.id.***layout\_payment***);  
 LinearLayout layoutCartNoItems = (LinearLayout) findViewById(R.id.***layout\_cart\_empty***);  
  
 **if**(MainActivity.*notificationCountCart* >0){  
 layoutCartNoItems.setVisibility(View.***GONE***);  
 layoutCartItems.setVisibility(View.***VISIBLE***);  
 layoutCartPayments.setVisibility(View.***VISIBLE***);  
 }**else** {  
 layoutCartNoItems.setVisibility(View.***VISIBLE***);  
 layoutCartItems.setVisibility(View.***GONE***);  
 layoutCartPayments.setVisibility(View.***GONE***);  
  
 Button bStartShopping = (Button) findViewById(R.id.***bAddNew***);  
 bStartShopping.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 finish();  
 }  
 });  
 }  
  
  
  
  
  
  
  
 }  
}

8.b WishListActivity

**package** com.allandroidprojects.buildsmart.options;  
  
**import** android.content.Context;  
**import** android.content.Intent;  
**import** android.net.Uri;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.support.v7.widget.LinearLayoutManager;  
**import** android.support.v7.widget.RecyclerView;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.ImageView;  
**import** android.widget.LinearLayout;  
  
**import** com.allandroidprojects.buildsmart.R;  
**import** com.allandroidprojects.buildsmart.product.ItemDetailsActivity;  
**import** com.allandroidprojects.buildsmart.utility.ImageUrlUtils;  
**import** com.facebook.drawee.view.SimpleDraweeView;  
  
**import** java.util.ArrayList;  
  
**import static** com.allandroidprojects.buildsmart.fragments.ImageListFragment.***STRING\_IMAGE\_POSITION***;  
**import static** com.allandroidprojects.buildsmart.fragments.ImageListFragment.***STRING\_IMAGE\_URI***;  
  
**public class** WishlistActivity **extends** AppCompatActivity {  
 **private static** Context *mContext*;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***layout\_recylerview\_list***);  
 *mContext* = WishlistActivity.**this**;  
  
 ImageUrlUtils imageUrlUtils = **new** ImageUrlUtils();  
 ArrayList<String> wishlistImageUri =imageUrlUtils.getWishlistImageUri();  
 RecyclerView recyclerView = (RecyclerView)findViewById(R.id.***recyclerview***);  
 RecyclerView.LayoutManager recylerViewLayoutManager = **new** LinearLayoutManager(*mContext*);  
  
 recyclerView.setLayoutManager(recylerViewLayoutManager);  
 recyclerView.setAdapter(**new** SimpleStringRecyclerViewAdapter(recyclerView, wishlistImageUri));  
 }  
  
 **public static class** SimpleStringRecyclerViewAdapter  
 **extends** RecyclerView.Adapter<WishlistActivity.SimpleStringRecyclerViewAdapter.ViewHolder> {  
  
 **private** ArrayList<String> **mWishlistImageUri**;  
 **private** RecyclerView **mRecyclerView**;  
  
 **public static class** ViewHolder **extends** RecyclerView.ViewHolder {  
 **public final** View **mView**;  
 **public final** SimpleDraweeView **mImageView**;  
 **public final** LinearLayout **mLayoutItem**;  
 **public final** ImageView **mImageViewWishlist**;  
 **public** ViewHolder(View view) {  
 **super**(view);  
 **mView** = view;  
 **mImageView** = (SimpleDraweeView) view.findViewById(R.id.***image\_wishlist***);  
 **mLayoutItem** = (LinearLayout) view.findViewById(R.id.***layout\_item\_desc***);  
 **mImageViewWishlist** = (ImageView) view.findViewById(R.id.***ic\_wishlist***);  
 }  
 }  
  
 **public** SimpleStringRecyclerViewAdapter(RecyclerView recyclerView, ArrayList<String> wishlistImageUri) {  
 **mWishlistImageUri** = wishlistImageUri;  
 **mRecyclerView** = recyclerView;  
 }  
  
 @Override  
 **public** WishlistActivity.SimpleStringRecyclerViewAdapter.ViewHolder onCreateViewHolder(ViewGroup parent, **int** viewType) {  
 View view = LayoutInflater.*from*(parent.getContext()).inflate(R.layout.***layout\_wishlist\_item***, parent, **false**);  
 **return new** WishlistActivity.SimpleStringRecyclerViewAdapter.ViewHolder(view);  
 }  
  
 @Override  
 **public void** onViewRecycled(ViewHolder holder) {  
 **if** (holder.**mImageView**.getController() != **null**) {  
 holder.**mImageView**.getController().onDetach();  
 }  
 **if** (holder.**mImageView**.getTopLevelDrawable() != **null**) {  
 holder.**mImageView**.getTopLevelDrawable().setCallback(**null**);  
*// ((BitmapDrawable) holder.mImageView.getTopLevelDrawable()).getBitmap().recycle();* }  
 }  
  
 @Override  
 **public void** onBindViewHolder(**final** WishlistActivity.SimpleStringRecyclerViewAdapter.ViewHolder holder, **final int** position) {  
 **final** Uri uri = Uri.*parse*(**mWishlistImageUri**.get(position));  
 holder.**mImageView**.setImageURI(uri);  
 holder.**mLayoutItem**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent intent = **new** Intent(*mContext*, ItemDetailsActivity.**class**);  
 intent.putExtra(***STRING\_IMAGE\_URI***,**mWishlistImageUri**.get(position));  
 intent.putExtra(***STRING\_IMAGE\_POSITION***, position);  
 *mContext*.startActivity(intent);  
 }  
 });  
  
 *//Set click action for wishlist* holder.**mImageViewWishlist**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 ImageUrlUtils imageUrlUtils = **new** ImageUrlUtils();  
 imageUrlUtils.removeWishlistImageUri(position);  
 notifyDataSetChanged();  
 }  
 });  
 }  
  
 @Override  
 **public int** getItemCount() {  
 **return mWishlistImageUri**.size();  
 }  
 }  
}

9.Payment Gateway [Payumoney]

9.a Constant

**package** com.allandroidprojects.buildsmart.payu\_de;  
  
  
  
**public class** Constant {  
  
 **public static final long *API\_CONNECTION\_TIMEOUT*** = 5000;  
 **public static final long *API\_READ\_TIMEOUT*** = 2000;  
  
 **public static final** String ***BASE\_URL*** = **"http://192.168.43.85/"**; *// https://www your domain .com/* **public static final** String ***SERVER\_main\_folder*** = **"payu/"**; *// use "foldername/" -- if www.yourdomain.com/foldername/app  
 // use "" -- if www.yourdomain.com/app  
//4012 0010 3714 1112  
 //05/20  
 //123*}

9.b MainActivity2.java

**package** com.allandroidprojects.buildsmart.payu\_de;  
  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.support.annotation.Nullable;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.view.View;  
**import** android.view.WindowManager;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
  
**import** com.allandroidprojects.buildsmart.R;  
  
  
**public class** MainActivity2 **extends** AppCompatActivity {  
  
 EditText **phone**, **amount**;  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 getWindow().setSoftInputMode(WindowManager.LayoutParams.***SOFT\_INPUT\_STATE\_HIDDEN***);  
 Button btn = (Button) findViewById(R.id.***start\_transaction***);  
 **phone** = (EditText) findViewById(R.id.***phone***);  
 **amount** = (EditText) findViewById(R.id.***amountid***);  
 btn.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 Intent intent = **new** Intent(MainActivity2.**this**, StartPaymentActivity.**class**);  
 intent.putExtra(**"phone"**, **phone**.getText().toString());  
 intent.putExtra(**"amount"**, **amount**.getText().toString());  
 startActivity(intent);  
 }  
 });  
 }  
}

9.c StartPaymentGateway

**package** com.allandroidprojects.buildsmart.payu\_de;  
  
**import** android.content.Intent;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.util.Log;  
**import** android.widget.Toast;  
  
**import** com.allandroidprojects.buildsmart.R;  
**import** com.payumoney.core.PayUmoneySdkInitializer;  
**import** com.payumoney.core.entity.TransactionResponse;  
**import** com.payumoney.sdkui.ui.utils.PayUmoneyFlowManager;  
  
**import** retrofit2.Call;  
**import** retrofit2.Callback;  
**import** retrofit2.Response;  
  
**public class** StartPaymentActivity **extends** AppCompatActivity {  
  
 PayUmoneySdkInitializer.PaymentParam.Builder **builder** = **new** PayUmoneySdkInitializer.PaymentParam.Builder();  
 *//declare paymentParam object* PayUmoneySdkInitializer.PaymentParam **paymentParam** = **null**;  
  
 String **TAG** =**"mainActivity"**, **txnid** =**"txt12346"**, **amount** =**"10"**, **phone** =**"9756423780"**,  
 **prodname** =**"Kuch bhi"**, **firstname** =**"Harish"**, **email** =**"harshitc889@gmail.com"**,  
 **merchantId** =**"12345"**, **merchantkey**=**"Wioja3s3"**; *// first test key only* @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_startpayment***);  
  
 Intent intent = getIntent();  
 **phone** = intent.getExtras().getString(**"phone"**);  
 **amount** = intent.getExtras().getString(**"amount"**);  
  
 startpay();  
 }  
  
 **public void** startpay(){  
  
 **builder**.setAmount(**amount**) *// Payment amount* .setTxnId(**txnid**) *// Transaction ID* .setPhone(**phone**) *// User Phone number* .setProductName(**prodname**) *// Product Name or description* .setFirstName(**firstname**) *// User First name* .setEmail(**email**) *// User Email ID* .setsUrl(**"https://www.payumoney.com/mobileapp/payumoney/success.php"**) *// Success URL (surl)* .setfUrl(**"https://www.payumoney.com/mobileapp/payumoney/failure.php"**) *//Failure URL (furl)* .setUdf1(**""**)  
 .setUdf2(**""**)  
 .setUdf3(**""**)  
 .setUdf4(**""**)  
 .setUdf5(**""**)  
 .setUdf6(**""**)  
 .setUdf7(**""**)  
 .setUdf8(**""**)  
 .setUdf9(**""**)  
 .setUdf10(**""**)  
 .setIsDebug(**true**) *// Integration environment - true (Debug)/ false(Production)* .setKey(**merchantkey**) *// Merchant key* .setMerchantId(**merchantId**);  
  
  
 **try** {  
 **paymentParam** = **builder**.build();  
 *// generateHashFromServer(paymentParam );* getHashkey();  
  
 } **catch** (Exception e) {  
 Log.*e*(**TAG**, **" error s "**+e.toString());  
 }  
  
 }  
  
 **public void** getHashkey(){  
 ServiceWrapper service = **new** ServiceWrapper(**null**);  
 Call<String> call = service.newHashCall(**merchantkey**, **txnid**, **amount**, **prodname**,  
 **firstname**, **email**);  
  
 call.enqueue(**new** Callback<String>() {  
 @Override  
 **public void** onResponse(Call<String> call, Response<String> response) {  
 Log.*e*(**TAG**, **"hash res "**+response.body());  
 String merchantHash= response.body();  
 **if** (merchantHash.isEmpty() || merchantHash.equals(**""**)) {  
 Toast.*makeText*(StartPaymentActivity.**this**, **"Could not generate hash"**, Toast.***LENGTH\_SHORT***).show();  
 Log.*e*(**TAG**, **"hash empty"**);  
 } **else** {  
 *// mPaymentParams.setMerchantHash(merchantHash);* **paymentParam**.setMerchantHash(merchantHash);  
PayUmoneyFlowManager.*startPayUMoneyFlow*(**paymentParam**, StartPaymentActivity.**this**, R.style.***AppTheme\_default***, **false**);  
 }  
 }  
  
 @Override  
 **public void** onFailure(Call<String> call, Throwable t) {  
 Log.*e*(**TAG**, **"hash error "**+ t.toString());  
 }  
 });  
  
 }  
  
 @Override  
 **protected void** onActivityResult(**int** requestCode, **int** resultCode, Intent data) {  
 **super**.onActivityResult(requestCode, resultCode, data);  
Log.*e*(**"StartPaymentActivity"**, **"request code "** + requestCode + **" resultcode "** + resultCode);  
 **if** (requestCode == PayUmoneyFlowManager.*REQUEST\_CODE\_PAYMENT* && resultCode == ***RESULT\_OK*** && data != **null**) {  
 TransactionResponse transactionResponse = data.getParcelableExtra( PayUmoneyFlowManager.*INTENT\_EXTRA\_TRANSACTION\_RESPONSE* );  
  
 **if** (transactionResponse != **null** && transactionResponse.getPayuResponse() != **null**) {  
  
 **if**(transactionResponse.getTransactionStatus().equals( TransactionResponse.TransactionStatus.***SUCCESSFUL*** )){  
 *//Success Transaction* } **else**{  
 *//Failure Transaction* }  
  
 *// Response from Payumoney* String payuResponse = transactionResponse.getPayuResponse();  
  
 *// Response from SURl and FURL* String merchantResponse = transactionResponse.getTransactionDetails();  
 Log.*e*(**TAG**, **"tran "**+payuResponse+**"---"**+ merchantResponse);  
 } }  
 }  
  
  
}

10 build.gradle (Module App)

apply **plugin**: **'com.android.application'**apply **plugin**: **'com.google.gms.google-services'**android {  
 compileSdkVersion 27  
 defaultConfig {  
 applicationId **"com.allandroidprojects.buildsmart"** minSdkVersion 21  
 targetSdkVersion 27  
 versionCode 1  
 versionName **"1.0"** testInstrumentationRunner **"android.support.test.runner.AndroidJUnitRunner"** }  
 buildTypes {  
 release {  
 minifyEnabled **false** proguardFiles getDefaultProguardFile(**'proguard-android.txt'**), **'proguard-rules.pro'** }  
 }  
}  
  
dependencies {  
 implementation fileTree(**dir**: **'libs'**, **include**: [**'\*.jar'**])  
 implementation **'com.google.firebase:firebase-database:16.0.1'** implementation **'com.google.firebase:firebase-storage:16.0.1'** implementation **'com.google.firebase:firebase-auth:16.0.1'** implementation **'com.google.firebase:firebase-core:16.0.1'** androidTestImplementation(**'com.android.support.test.espresso:espresso-core:2.2.2'**, {  
 exclude **group**: **'com.android.support'**, **module**: **'support-annotations'** })  
 api **'com.android.support:appcompat-v7:27.1.1'** api **'com.facebook.android:facebook-android-sdk:[4,5)'** implementation **'com.android.support:design:27.1.1'** testImplementation **'junit:junit:4.12'** api **'com.android.support:cardview-v7:27.1.1'** api **'com.android.support:recyclerview-v7:27.1.1'** api **'com.facebook.fresco:fresco:0.14.1'** api **'com.github.JakeWharton:ViewPagerIndicator:2.4.1'** implementation **'com.android.support.constraint:constraint-layout:1.1.3'** implementation **'com.android.support:customtabs:27.1.1'** implementation **'com.android.support:animated-vector-drawable:27.1.1'** *//payumoney* implementation **'com.payumoney.core:payumoney-sdk:7.4.4'** implementation **'com.payumoney.sdkui:plug-n-play:1.5.0'** *// retrofit library* implementation **'com.squareup.retrofit2:retrofit:2.5.0'** implementation **'com.squareup.retrofit2:converter-gson:2.3.0'** implementation **'com.squareup.okhttp3:logging-interceptor:3.4.1'** implementation **'com.squareup.okhttp3:okhttp:3.4.1'**

implementation **'com.google.firebase:firebase-ml-vision:15.0.0'**

}  
com.google.gms.googleservices.GoogleServicesPlugin.config.disableVersionCheck = **true**

11.Searching by detecting text in image

**package** com.example.ashwin.textdetector;  
  
**import** android.content.Intent;  
**import** android.graphics.Bitmap;  
**import** android.provider.MediaStore;  
**import** android.support.annotation.NonNull;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.ImageView;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** com.google.android.gms.tasks.OnFailureListener;  
**import** com.google.android.gms.tasks.OnSuccessListener;  
**import** com.google.firebase.ml.vision.FirebaseVision;  
**import** com.google.firebase.ml.vision.common.FirebaseVisionImage;  
**import** com.google.firebase.ml.vision.text.FirebaseVisionText;  
**import** com.google.firebase.ml.vision.text.FirebaseVisionTextDetector;  
  
**import** java.util.List;  
  
**public class** MainActivity **extends** AppCompatActivity {  
  
 **private** Button **snapBtn**;  
 **private** Button **detectBtn**;  
 **private** ImageView **imageView**;  
 **private** TextView **txtView**;  
 **private** Bitmap **imageBitmap**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 **snapBtn** = findViewById(R.id.***snapBtn***);  
 **detectBtn** = findViewById(R.id.***detectBtn***);  
 **imageView** = findViewById(R.id.***imageView***);  
 **txtView** = findViewById(R.id.***txtView***);  
 **snapBtn**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 dispatchTakePictureIntent();  
 }  
 });  
 **detectBtn**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 detectTxt();  
 }  
 });  
 }  
  
 **static final int *REQUEST\_IMAGE\_CAPTURE*** = 1;  
  
 **private void** dispatchTakePictureIntent() {  
 Intent takePictureIntent = **new** Intent(MediaStore.***ACTION\_IMAGE\_CAPTURE***);  
 **if** (takePictureIntent.resolveActivity(getPackageManager()) != **null**) {  
 startActivityForResult(takePictureIntent, ***REQUEST\_IMAGE\_CAPTURE***);  
 }  
 }  
  
 @Override  
 **protected void** onActivityResult(**int** requestCode, **int** resultCode, Intent data) {  
 **if** (requestCode == ***REQUEST\_IMAGE\_CAPTURE*** && resultCode == ***RESULT\_OK***) {  
 Bundle extras = data.getExtras();  
 **imageBitmap** = (Bitmap) extras.get(**"data"**);  
 **imageView**.setImageBitmap(**imageBitmap**);  
 }  
 }  
  
 **private void** detectTxt() {  
 FirebaseVisionImage image = FirebaseVisionImage.*fromBitmap*(**imageBitmap**);  
 FirebaseVisionTextDetector detector = FirebaseVision.*getInstance*().getVisionTextDetector();  
 detector.detectInImage(image).addOnSuccessListener(**new** OnSuccessListener<FirebaseVisionText>() {  
 @Override  
 **public void** onSuccess(FirebaseVisionText firebaseVisionText) {  
 processTxt(firebaseVisionText);  
 }  
 }).addOnFailureListener(**new** OnFailureListener() {  
 @Override  
 **public void** onFailure(@NonNull Exception e) {  
  
 }  
 });  
 }  
  
 **private void** processTxt(FirebaseVisionText text) {  
 List<FirebaseVisionText.Block> blocks = text.getBlocks();  
 **if** (blocks.size() == 0) {  
 Toast.*makeText*(MainActivity.**this**, **"No Text :("**, Toast.***LENGTH\_LONG***).show();  
 **return**;  
 }  
 **for** (FirebaseVisionText.Block block : text.getBlocks()) {  
 String txt = block.getText();  
 **txtView**.setTextSize(24);  
 **txtView**.setText(txt);  
 }  
 }  
  
}

**REFERENCE**

1 Mewer R. Professional Android 2 Application Development. Indianapolis: Wiley Publishing,

Inc.; 2010.

2 Göransson A. Efficient Android Threading. Gravenstein Highway North, Sebastopol,

CA: O’Reilly Media, Inc.; 2014.

3 Android-App. Android architecture–The Key Concepts of Android OS [serial online].

February 2012.

URL: http://www.android-app-market.com/android-architecture.html.

Accessed December 11, 2014.

4 Android Open Source Project. Application Fundamentals – App Components [online].

Google; May 2014.

URL: http://developer.android.com/guide/components/fundamentals.html.

Accessed December 11, 2014.

5 PayUMoney. Introducing Adaptive Payments [online]. PayUMoney Inc.; August 12, 2014.

adaptive-payments/integration-guide/APIntro.

Accessed December 12, 2014.

6 Lim J. Algorithms, Flowcharts, Data Types and Pseudo Code [online]. CA: scribed;

January 2011. URL: http://www.scribd.com/doc/46981114/Algorithms-Flowcharts-Data-

Types-and-Pseudo-Code. Accessed 13 November 2014.

7 Google. Android 2.3 Compatibility Definition [online]. Google Inc; 2010.

URL: http://static.googleusercontent.com/media/source.android.com/

en//compatibility/2.3/android-2.3-cdd.pdf

Accessed December 12, 2014.