**E-money**

From: Ramya Radhakrishnakumar, Sushant Sharma and Vinokkumar Uthayakumar  
Discipline: Computer Engineering Technology  
Date: April 22, 2018

# Declaration of Joint Authorship

We acknowledge that this report to be assessed is a group work by Ramya Radhakrishnakumar, Sushant Sharma and Vinokkumar Uthayakumar. The entire report and the project itself is prepared with the group’s consent and has almost our own ideas and words. All the other references and ideas invoked in this report are confirmed and are provided in the bibliography session. Ramya Radhakrishnakumar worked on the Hardware and the Technical aspect, Sushant Sharma handled the Database connectivity and Vinokkumar Uthayakumar on the other hand focused on developing the mobile application and the establishing connectivity between the sensors.

# Approved Proposal

**Executive Summary**

As a student in the Computer Engineering Technology program, I will be integrating the knowledge and skills I have learned from our program into this Internet of Things themed capstone project. This proposal requests the approval to build the hardware portion that will connect to a database as well as to a mobile device application. The internet connected hardware will include a custom PCB with the following sensors and actuators user login/sign up information, fingerprint data and Banking account information. The database will store #REF!. The mobile device functionality will include It enables us to handle our banking needs and also lets us generate the QR-code for the money we intend to send/transfer/pay. and will be further detailed in the mobile application proposal. I will be collaborating with the following company/department Humber college parts crib, prototype laboratory, Humber library, toronto public liberary and A1 electronics.. In the winter semester I plan to form a group with the following students, who are also building similar hardware this term and working on the mobile application with me Ramya Radhakrishnakumar, Vinokkumar Uthayakumar and Sushant Sharma.. The hardware will be completed in CENG 317 Hardware Production Techniques independently and the application will be completed in CENG 319 Software Project. These will be integrated together in the subsequent term in CENG 355 Computer Systems Project as a member of a 2 or 3 student group.

**Background**

The problem solved by this project is Certain banks have limitations such as one can use his/her bank card for free only for certain number of transactions in a month. If someone has to use more than the limited transaction allotted in a month then he/she might be charged an interest. To avoid this problem, we have come up with an idea which generates a QR code for the money one wants to transfer.. A bit of background about this topic is We thought of E-Money transfer application because we wanted to reduce the use of card to do transaction or going to a bank to send money to a person/a merchandise whom we owe money. We wanted to enhance the security when transferring money via the internet. We have decided to use fingerprint reader in our device to fulfill security requirement. This project is better than money transfer online through website because one just needs a QR scanner and a fingerprint reader to accept the money rather an answer for a security question. Using the QR code makes the transfer quicker compare to online transfer. By generating the QR code the recipient scans the code. This displays a small information containing who sent the money, who will be receiving it and the amount being sent. Doing so will reduce multiple payments to the same recipient. Now the person who receives the money can login to his account after authenticating access using his fingerprint to check if the money was deposited in his account..

Existing products on the market include [1]. I have searched for prior art via Humber’s IEEE subscription selecting “My Subscribed Content”[2] and have found and read [3] which provides insight into similar efforts.

In the Computer Engineering Technology program, we have learned about the following topics from the respective relevant courses:

* Java Docs from CENG 212 Programming Techniques in Java,
* Construction of circuits from CENG 215 Digital and Interfacing Systems,
* Rapid application development and Gantt charts from CENG 216 Intro to Software Engineering,
* Micro computing from CENG 252 Embedded Systems,
* SQL from CENG 254 Database with Java,
* Web access of databases from CENG 256 Internet Scripting; and,
* Wireless protocols such as 802.11 from TECH152 Telecom Networks.

This knowledge and skill set will enable me to build the subsystems and integrate them together as my capstone project.

**Methodology**

This proposal is assigned in the first week of class and is due at the beginning of class in the second week of the fall semester. My coursework will focus on the first two of the 3 phases of this project:  
 Phase 1 Hardware build.  
 Phase 2 System integration.  
 Phase 3 Demonstration to future employers.

*Phase 1 Hardware build*

The hardware build will be completed in the fall term. It will fit within the CENG Project maximum dimensions of 12 13/16" x 6" x 2 7/8" (32.5cm x 15.25cm x 7.25cm) which represents the space below the tray in the parts kit. The highest AC voltage that will be used is 16Vrms from a wall adaptor from which +/- 15V or as high as 45 VDC can be obtained. Maximum power consumption will be 20 Watts.

*Phase 2 System integration*

The system integration will be completed in the fall term.

*Phase 3 Demonstration to future employers*

This project will showcase the knowledge and skills that I have learned to potential employers.

The brief description below provides rough effort and non-labour estimates respectively for each phase. A Gantt chart will be added by week 3 to provide more project schedule details and a more complete budget will be added by week 4. It is important to start tasks as soon as possible to be able to meet deadlines.

We are planning to purchase jumper wires, an acrylic case for sensors and raspberry, android phone, breadboard, led display, data cable wire to connect the phone to the pi. Wires to connect all the sensors together. And also need to purchase the firebase(database) membership.

**Concluding remarks**

This proposal presents a plan for providing an IoT solution for We will be working together consistently as a team every week. We will be getting help from Mr.Kelly and Mr.Vlad to get better understanding to things. Safety is our first priority and we will make sure that we do not take any dangerous step. Also we would like to work on the software part so that it collaborates with our hardware needs.. This is an opportunity to integrate the knowledge and skills developed in our program to create a collaborative IoT capstone project demonstrating my ability to learn how to support projects such as the initiative described by [3]. I request approval of this project.

# 

# Abstract

The peculiarity of the mobile application is that one can transfer, pay and also manage their bank account all in one. It is something like a banking application but has more features to it such as one does not have the need to go to a bank to transfer money to someone far away also the user can pay merchandised bills using this application. E-money application needs sufficient amount of memory and access to the cellular data/Wi-Fi. The ideology of this project is to save time, carry transactions and merchandize bill payments with ease. Also, this application can prevent investing money for purchasing credit cards. This application is available for android mobile OS only for time being.

Table of Contents

[Declaration of Joint Authorship iii](#_Toc512253244)

[Approved Proposal iv](#_Toc512253245)

[Abstract x](#_Toc512253246)

[1. Introduction 14](#_Toc512253247)

[2. Project Description 15](#_Toc512253248)

[2.1 Problem 15](#_Toc512253249)

[2.2 Rationale Behind Project 15](#_Toc512253250)

[2.3 Project scope 15](#_Toc512253251)

[2.4 Software Requirement Specifications 16](#_Toc512253252)

[2.4.1 Purpose 16](#_Toc512253253)

[2.4.2 Product Perspective 17](#_Toc512253254)

[2.4.3 Product Functions 18](#_Toc512253255)

[2.4.4 User Classes and Characteristics 18](#_Toc512253256)

[2.4.5 Operating Environment 19](#_Toc512253257)

[2.4.6 Design and Implementation Constraints 19](#_Toc512253258)

[3 External Interface Requirements 20](#_Toc512253259)

[3.1 Hardware Interfaces 20](#_Toc512253260)

[3.2 Software Interfaces 20](#_Toc512253261)

[4. Mobile Application Features 20](#_Toc512253262)

[4.1 Case Scenarios 20](#_Toc512253263)

[4.2 Database 21](#_Toc512253264)

[Database 1: (Bank account) 23](#_Toc512253265)

[Database 2: (Merchant) 24](#_Toc512253266)

[4.3 Mobile Application 24](#_Toc512253267)

[5. Project Overview 25](#_Toc512253268)

[5.1 Fingerprint Sensor 25](#_Toc512253269)

[5.1.1 Introduction 25](#_Toc512253270)

[5.1.2 Invoice/Bill: 26](#_Toc512253271)

[5.1.3 Budget 27](#_Toc512253272)

[5.1.4 Time Commitment 27](#_Toc512253273)

[5.1.5 Mechanical Assembly of my Fingerprint sensor 27](#_Toc512253274)

[5.1.6 Power up 28](#_Toc512253275)

[5.1.7 Installation of the Raspberry Pi Fingerprint Library 29](#_Toc512253276)

[5.1.8 Unit Testing 29](#_Toc512253277)

[5.1.9 Production testing 30](#_Toc512253278)

[5.2 QR Scanner 30](#_Toc512253279)

[5.2.1 Introduction 30](#_Toc512253280)

[5.2.2 Bill of Materials Budget 31](#_Toc512253281)

[5.2.3 Time Commitment 31](#_Toc512253282)

[5.2.4 Mechanical Assembly 32](#_Toc512253283)

[5.2.5 Unit Testing 32](#_Toc512253284)

[5.2.6 Production Testing 32](#_Toc512253285)

[6. Problems and Approaches 33](#_Toc512253286)

[6.1 Data Communication 33](#_Toc512253287)

[6.2 LCD Display 33](#_Toc512253288)

[6.3 Mobile Application 34](#_Toc512253289)

[6.4 QR code email 34](#_Toc512253290)

[7. Progress Reports 35](#_Toc512253291)

[7.1 Report 1 35](#_Toc512253292)

[7.2 Report 2 38](#_Toc512253293)

[7.3 Report 3 40](#_Toc512253294)

[8. Conclusions 42](#_Toc512253295)

[9. Recommendations 42](#_Toc512253296)

[10. Technical References/Bibliography 42](#_Toc512253297)

[11. Appendices 43](#_Toc512253298)

# Introduction

A significant problem being faced by customers these days is that, every customer has to go to the bank, wait in line to be helped and then consult our needs with the representative to perform his/her banking needs such as monitoring the bank account, transfer money, pay bills, security and credit card fraud. The e-money application can overcome these problems.

To make this application work, it requires certain information such as Acc.no, name and password to be stored in the database which will then be accessed via both a hardware and Smartphone application, providing on-the-go banking experience. This application is convenient and efficient keeping user friendliness as a key to work with the application.

Since our mobile application needs a Wi-Fi or cellular data to send an email to the recipient, the whole set-up should be placed in an environment that supports either Wi-Fi or cellular data.

# 

# Project Description

### Problem

The problem encountered is, the user has to go to a bank to meet their daily banking needs such as transfer money, managing bank account and also pay merchandiser bills. The most current existing mobile applications do not have all the above features and it seems as though the user need to download more than one mobile application. This intern leads to large memory consumption and security issues. There are few banking mobile applications but these have few limitations. One of the major problem is that if the user enters the wrong account number followed by the wrong routing number it will cause the money to go into another account. Also, the user needs to double check the amount the user intends to send.

### Rationale Behind Project

In order to solve the above-mentioned problems, the team came up with the idea of E-money and worked towards this project. It is simply an inexpensive, user friendly, very specific, sophisticated and realistic mobile application being named the E-money. From the very name the mobile application carries all its functions using the internet. This mobile application will be the key to security for your banking information and also it is free of charge. It helps in transferring, paying and also managing accounts from one place.

### 2.3 Project scope

E-Money is so beneficial since the users need to have to waste time going into a physical bank and wait for a teller to help them out to transfer money. This can also be out stretched and used to pay the application user’s mortgage or car loan. This is much quicker as a transfer processed by the bank takes a minimum of 2 business days to transfer any amount to any account. It is something like a direct deposit, here the user need not be physically present in the bank to perform the deposit. The security is taken over by the fingerprint reader. This simply accesses the bank database and matches the finger print to the email ID along with the card number and then pulls out the banking information. By doing so, there won’t be any miscellaneous handle of banking data’s and only the authorised user can only handle the information.

Since the intention is to produce an user-friendly and inexpensive mobile application, some of the high level security features and also banking information has been voided. Like, the user can register by using the email ID and the intended finger print. But the user need not provide the card number of the card that is issued by the bank. Also the user need not prove other personal details such as photo ID, home address, contact number, etc. . Likewise, any user can register into this application regardless of having an account in that particular bank, since that requires a well maintained banking database which keeps updating as soon as bank card is issued or expired. It also provides access to the QR code without the need to enter a passcode to open the file sent via email to transfer money to the recipient.

## 2.4 Software Requirement Specifications

## 2.4.1 Purpose

The hardware products which are going to be included in our project are fingerprint sensor and QR code scanner. The software products are firebase and android studios. The whole system will be built upon the basis of these products. This system can be further divided in two subsystems such as hardware and software aspects. The purpose for this project is to decrease the use of cash or credit/debit cards and use the money generated via QR code. The reason for using QR code is that we don’t have to worry about losing cash or even if we forget our wallet at home we can use our cell phone to pay with QR code.

## 2.4.2 Product Perspective

This product is basically a replacement of the existing banking applications and apple pay/android pay. This is built keeping security as the primary key. The application is designed in a way that the user can perform or manage transactions as well as pay their bills all in one. The following schematic gives a clear understanding:

**Case 1 & 2:**

**E-money application**

* Transfer money
* Pay using QR scanner

DB

Generate the code and let the merchandiser scan it

Generate QR code and send the code through an email to the recipient

Payment successful ☺

QR scanner scans the code and deposits the money

**Case 3:**

Enter Acc.no: 1234567890

Account Balance: $3250.75

Credit: $200

Scan

Finger Print sensor

## Product Functions

\* The Code scanner is used to scan the generated code.

\* Mobile application is used to generate the QR code.

\* The Fingerprint sensor is used for security purpose.

## 2.4.4 User Classes and Characteristics

Almost 90% of world’s population has their day starting with an android smart phone. This project is intended for people having access to their bank account and smartphone. In this era everything is possible via a smartphone for instance, one can order food, taxi, etc. using various android apps. According to e-money the use may vary from an employer to a merchandiser. This application is user friendly for both and is designed in such a way that both has access to all facilities provided by the app itself. People use online banking since one can manage monthly statements and perform transactions with ease. Whereas the merchandiser has to create a default account so that the payment made to them by the customer gets deposited in it.

## 2.4.5 Operating Environment

The environment such as weather, location etc., is not an asset for the e-money. The whole hardware is not legitimately big and hence it is easy to carry around in the parts kit itself. The application is intended for android operating system and hence it is not available for other mobile operating systems. The application collides with the fingerprint sensor to give permissions to access the user’s banking account. The barcode scanner plays a part along with the software application to perform transactions. This is how it is being organized to make the hardware and software work together.

## 2.4.6 Design and Implementation Constraints

The database incorporated in our product contains bank account info such as the ten-digit account number, account types like chequing account and savings account. The database will also include information about user login which will have the username as the account/card number, the password field as well as fingerprint for enhanced security. If the user does not have an account then they can register and make an account in order to login. For registering the user will be asked to enter their first name, last name, email id, cell phone number, account number/username, fingerprint and password. All this information will the stored in database and will be needed in order to login.

# 3 External Interface Requirements

## 3.1 Hardware Interfaces

The hardware interfaces needed for connecting our components differ from sensor to sensor. QR code scanner is going to be connected using a USB interface with the raspberry pi. The fingerprint scanner is connected through the TTL to USB convertor with the pi.

## 3.2 Software Interfaces

The database incorporated is firebase and the operating system used is Android studios (lollipop 22). The mobile application was conveniently developed with the help of Android studios which uses tools and libraries from the internet. The services needed are an internet connection, access to email protocol and converting QR code to image file and finger print sensor. Also some python code was incorporated to showcase a presentable GUI which will ease user interface.

# 4. Mobile Application Features

## 4.1 Case Scenarios

**Case 1: e-mail Transfer:**

In the user interface there is an option called send. Once send is selected it prompt the user to enter the amount to which we want to transfer. This then generated the QR code with the Acc.no and amount. This is packaged into a unique number. This is used only for transferring.

Now the recipient who receives the email with the unique number will be able to deposit the money if he takes the barcode to the bank. The bank who has the barcode scanner scans the code and deposits the money to the recipient’s account.

**Case 2: Payment:**

In this case the user will have access to generate the QR code for the amount he owes the merchandiser. This is packaged into the unique number (varies from the 1st case). This code is scanned by the merchandiser. Now the amount gets deposited directly to the merchandiser’s default account.

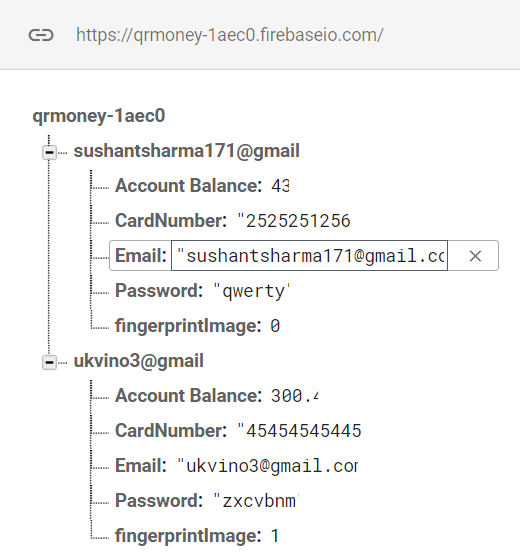
**Case 3: Account Detail:**

Now if the user wants to check the Account information he has to enter his Acc.no and scan the fingerprint associated with the bank account. The fingerprint is basically used for security purposes.

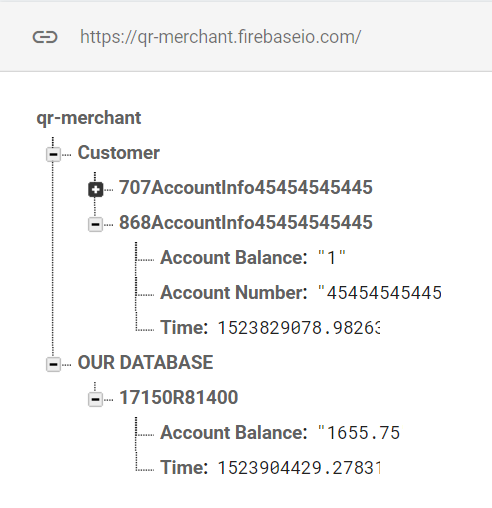
### 4.2 Database

The parent database is controlled by the bank which has details about the users such as name, card number, account balance, transaction history along with the fingerprint enrolled with the application. Also, we have incorporated two separate child database for the need of the mobile application. One database is for the mobile application user where the login credentials such as Acc. Balance, card number, email, password and fingerprint of that particular user is used to enroll or register with the application. Only if the information matches it allows to access the account. If it does not match it won’t allow permission to access the account. The second database contains the merchant (Tim-Hortons, Subway. etc) database. This is like a merchant’s account where information such as Acc. Balance of the merchant is available. Along with the account balance there is some additional information such as payment details of the customer and also the time stamp of when and which location the payment was performed. The time stamp is implicated in both the child process because it can be easy to check any miscellaneous payment or security issues

### Database 1: (Bank account)

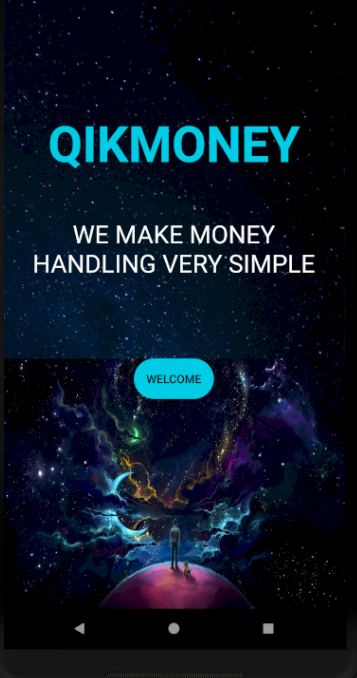
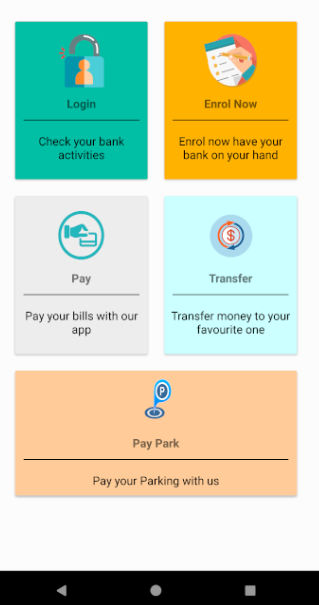


### Database 2: (Merchant)



### 4.3 Mobile Application

In the application we added a navigation bar that uses fragments to switch between each layout between screens. Some of the methods/functions added are overview, pay, transfer, login and enroll. Every option has a short description of what it leads to by selecting it. We also added layouts for each of those methods that are mentioned above. We added customized draw able images and icons so that it looks more interactive. Some libraries are used for the firebase connectivity with the application which retrieves information from the database.

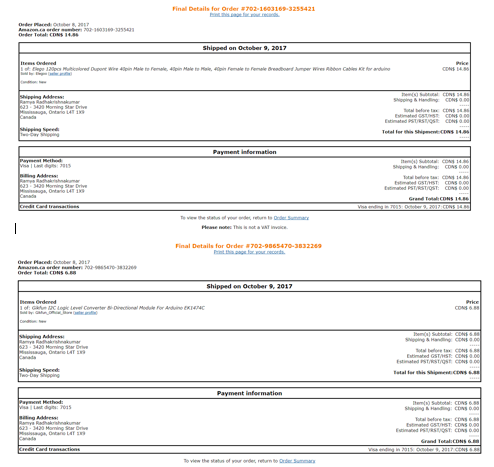
# 5. Project Overview

## 5.1 Fingerprint Sensor

## 5.1.1 Introduction

E-money transfer is a convenient way to send, receive or pay bills in store using a simple QR code generator. We are using a finger print sensor in collaboration with the software app. The sensor is used as a safety measure to prevent loss of information. In order to access your account, you need to login using your login credentials including the finger print pattern that is registered along with your bank account in the E-money application. Basically, the finger print lets you enroll, delete, search or generate a picture of your finger. To do this we used a raspberry pi along with a usb to ttl convertor and jumper wires.

## 5.1.2 Invoice/Bill:



Ordered the sensor from Adafruit website. Also ordered raspberry pi, USB to TTL and jumper wires from Amazon. It roughly took a week to reach.

### 5.1.3 Budget

* The Fingerprint sensor (751) costed $64.00.
* The Raspberry pi costed $99.99.
* The USB to TTL convertor costed $6.99.
* The Jumper wires costed $5.99

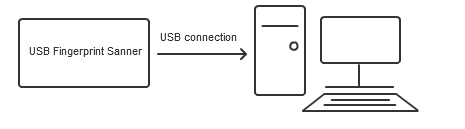
## 5.1.4 Time Commitment

The time commitment was a cake walk since I created a schedule during the start of my semester.

It really was one reason that we organized the work properly and was able to achieve the milestones as intended.

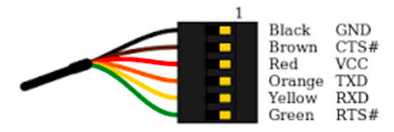
Setting up the hardware must take around 20 minutes provided if you have the equipment’s and have previous knowledge of the hardware. We took roughly 1 week to set it up. We used python code to make the sensor work. The testing should take around 10 minutes, but actually took time to read the code and understand what was actually in the code. It took 3 days to work on understanding the code and then successfully executed.

## 5.1.5 Mechanical Assembly of my Fingerprint sensor

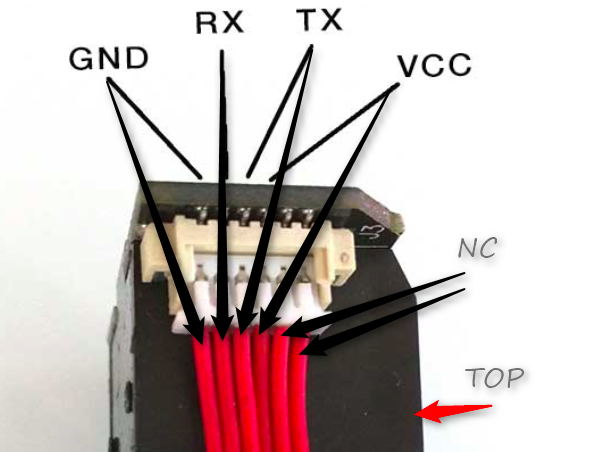


1. Connect the fingerprint to the USB to TTL convertor as follows:

2. This is the pinout of the USB to TTL.



3. The pinout of the fingerprint sensor.



4. Connections using the jumper wires:

\*Connect the GN of the sensor to the GN of the TTL converter.

\*Connect the RX of the sensor to the TX of the TTL converter.

\*Connect the TX of the sensor to the RX of the TTL converter.

\*Connect the VCC of the sensor to the VCC of the TTL converter.

5. Then connect the sensor through the USB port of the Raspberry Pi.

## 5.1.6 Power up

So once assembled the hardware part, then connected my mouse and keyboard along with the charger cable to my pi and then turned it ON. Also we need not connect the Ethernet cable since we already configured the pi to connect it to the Wi-Fi. We were able to see the sensor light up. You will be able to achieve this if your hardware connection is perfect.

## 5.1.7 Installation of the Raspberry Pi Fingerprint Library

We need to go to the root for some commands of the installation. Then we started a terminal session and typed the following:

sudo bash

Now add the necessary package sources:

wget -O - http://apt.pm-codeworks.de/pm-codeworks.de.gpg | apt-key add -

wget http://apt.pm-codeworks.de/pm-codeworks.list -P /etc/apt/sources.list.d/

Then had to update the available packages and install the Python library as follows:

apt-get update

apt-get install python-fingerprint –yes

Now the packages are installed and ready for testing.

## 5.1.8 Unit Testing

1. As mentioned above the package has files for storing a new fingerprint, reading out and deleting stored fingerprints. Now we tested the enrolling option by issuing the following:

python2 /usr/share/doc/python-fingerprint/examples/example\_enroll.py

2. The terminal asks you to place the finger you want to enroll on the sensor, wait for the message "remove finger" and then again "place the same finger".

Now the finger is given an ID and it is enrolled.

3. Now we test if the finger is recognized. So we issued the following script:

python2 /usr/share/doc/python-fingerprint/examples/example\_search.py

4. Then put the finger on it again. If the fingerprint on the Raspberry Pi is detected, a message like this appears:

Currently stored templates: 2

Waiting for finger...

Found template at position #1

The accuracy score is: 90

SHA-2 hash of template: 3aa1b01149abf0a7ad0d7803eaba65c22ba084009700c3c7f5f4ecc38f020851

## 5.1.9 Production testing

The testing was successful. We were was to make the sensor work by enrolling the finger, delete the finger by giving the appropriate ID, search for a fingerprint that was enrolled and retrieve the fingerprint image. This is a very reliable for security purposes.

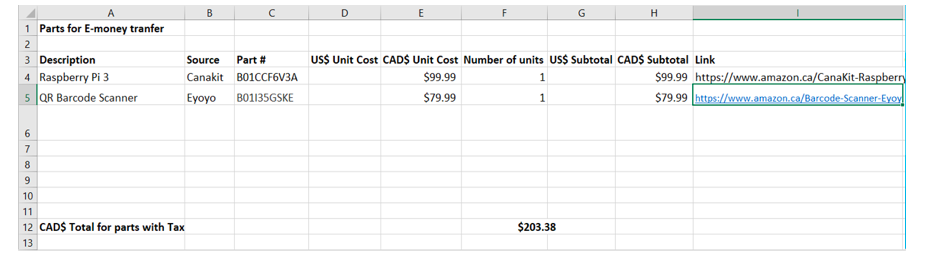
## 5.2 QR Scanner

## 5.2.1 Introduction

This project is about e-money transfer by using Qr code sensor. The components that have been used for this project are raspberry pi 3 and Qr code sensor. The raspberry pi comes with a case, power adapter and micro sd card with raspbian OS. A monitor, mouse and a keyboard is required to operate raspberry pi, unless you have laptop to connect it with. To connect a laptop with the pi, you have to configure specific settings accordingly which is not covered in the following instructions. The qr code sensor is connected through a usb with raspberry pi. Recommended libraries have been used to power up the sensor. I plan to demonstrate transactions of an amount of money using the qr code sensor. It can be used in stores to buy stuff and to transfer money. Building this project should require a couple of hours if you follow these instructions.

## 5.2.2 Bill of Materials Budget

We ordered raspberry pi from amazon and it arrived within three days as we used amazon prime shipping. For the most part, we used QR code scanner and raspberry pi. QR code sensor can be used to scan both barcodes and QR codes which is one of the reasons they are more expensive compared to barcode scanners. They can also scan both 1 dimensional and 2 dimensional codes and are equipped with laser.

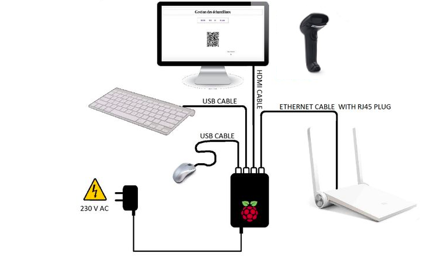


## 5.2.3 Time Commitment

Once we received the equipment, it took approximately 2 weeks to complete the project. This period includes the time that we spent on researching and building the sensors in order for them to operate in the way we wanted them to. The QR code scanner is connected to the pi through a USB interface which took about 2 minutes to connect and test. However, the part that took the most amount of the time was creating the python code.

## 5.2.4 Mechanical Assembly

We did not need any extra electrical equipment such as wires, motors, resistors, capacitors etc. The assembly of this project requires raspberry pi, power adaptor and QR code scanner with a USB interface. Just used a mouse, keyboard and monitor to connect the pi and to test my sensor.



## 5.2.5 Unit Testing

When we first received the sensor, we were testing it by connecting it to my powered up pi. Initially, was just scanning randomly generated QR codes and barcodes from the internet. And every time we scanned, the screen we were on, happened to scroll down automatically. So, thought that the scanned item was being stored somewhere on the SD card of the pi. However, the attempt was unsuccessful. Then thought that maybe it would be stored in a text editor like notepad or MS word. So we scanned while opening a text editor and the code was stored in the file.

## 5.2.6 Production Testing

Then we developed the Python code for the intended function to be carried by the sensor. The developed code works in a way that it is connected to the firebase database account. Once the QR code generated by the application is scanned it will recognize the amount that was stored in the QR code and it will add that amount to the linked bank account in the firebase database.

# 6. Problems and Approaches

### 6.1 Data Communication

The mobile application has many fragments. One of the fragment is used to get the user information. This asks the user to enter the card number, email ID and fingerprint. The issue was the application being connected to the Firebase database was not able to fetch the account information despite giving the correct credentials.

Also issue with the python code is something about the version. It is an in built function of the firebase which may not compile the python code if the version of the python code and firebase is outdated. The python code will not be issued or compiled resulting in not displaying the GUI. So it is always wiser to have the same working version for both the firebase and the python code to avoid such minor errors. This intern leads to not fetching the data stored in the firebase database which is a serious issue. Once the version was changed, the python code was able to access the firebase database and present the GUI.

### 6.2 LCD Display

A Liquid Crystal Display was incorporated in the project. The LCD’s purpose was to present the GUI of the project. But the issue was that, it wasn’t turning on. But the touch of the screen happened to work. The team then came up with a conclusion that it was a hardware issue. After certain encounters and learning phases the problem was with the adapter of the LCD. The LCD required a minimum voltage of 5V but the adapter that was used only supplied 3.3V. Once the adapter was replaced, the LCD was up and running displaying the GUI once the python code was established.

### 6.3 Mobile Application

The mobile application had a difficulty which intern comes under software issues. The buttons and variables are used in the fragments. These buttons and variables couldn’t be passed to other fragments. This was the issue, one solution was to bundle the buttons or variables. The procedure is to bundle these buttons or variables into one and then pass this bundle as indented to other fragment. Another solution used is also similar to the previous one. Here the only difference is we bundle the object. And this bundled object will be passed to another fragment. The issue with the second solution is that it sometimes returns a null. The 1st solution is more reliable than the second.

### 6.4 QR code email

One of the case scenarios is that using the mobile application the user can send money to another user whom he owes. So to perform this transfer it requires the sender to generate the QR code and send that code over email. Initially here the issue was QR code was not transmitted via email to the receiver. It keeps asking the sender to save the QR code to the images folder of the phone. Only after saving the QR code in the image folder the user was able to email it. Then after numerous trouble shooting it was fixed by incorporating some libraries into the Android code. Now the QR code need not be saved into the image folder. It can be created and sent directly by creating the QR code for the indented transfer and entering the receiver’s email ID.

# 7. Progress Reports

## 7.1 Report 1

Ramya Radhakrishnakumar < [ramyarkkumar27@gmail.com](mailto:ramyarkkumar27@gmail.com) >

To: Austin Tian < [Austin.Tian@humber.ca](mailto:Austin.Tian@humber.ca) >

Cc: Sushant Sharma < [sushantsharma88888@gmail.com](mailto:sushantsharma88888@gmail.com) >, Vinokkumar Uthayakumar<[ukvino3@gmail.com](mailto:ukvino3@gmail.com) >

Dear Austin Tian,

We have decided student A to be Ramya Radhakrishnakumar, student B to be Sushant Sharma and student C to be Vinokkumar Uthayakumar. This email is to update the progress shown in e-money since week 3.

Since our very idea of the project has been overlooked, few changes had to be made to the mobile application. So we dedicated more attention towards it. Vino worked on the changes to be made in that application and some of the modifications are; we added a navigation bar that uses fragments to switch between each layout between screens. This actually gives a proper flow for the application and also made it a bit more user friendly now than before. Some of the methods/functions added are contact, enrolment, overview, pay and transfer. We also added layouts for each of those methods that are mentioned above. Moving on to the database part controlled by Sushant, the database has been created successfully and we tried to connect it to the mobile application. It wasn’t connecting initially because of some technical issue. We spent more time on fixing the issue and now we are able to connect the database to the mobile application. We are able to interact between the application and the database now. Ramya made sure if the hardware/sensors incorporated in this project are in proper working conditions. We are yet to work on the hardware and database connection this week. Also we carried out some research on the hardware parts that might be needed in the future.

So far there is no change in the finical status of the project. It remains the same. Further updates on the budget are to be seen down the following weeks.

 As per the project schedule we have completed almost as intended until week 5, yet the major milestone is yet to be completed and that is the connection of the database to the hardware (QR scanner and Fingerprint sensor) which we are planning to work on week 6 and also we might get help from Kelly in the prototype laboratory.

Links to the Media added on GitHub (<https://github.com/RamyaRadhakrishnakumar/ceng355>):

<https://github.com/RamyaRadhakrishnakumar/ceng355/blob/master/PastedGraphic-1.png>

<https://github.com/RamyaRadhakrishnakumar/ceng355/blob/master/PastedGraphic-2.png>

<https://github.com/RamyaRadhakrishnakumar/ceng355/blob/master/Screen%20Shot%202018-03-05%20at%202.04.40%20PM.png>

Some of the websites used as reference for the application and database development are as follows:

<https://www.youtube.com/watch?v=F6UWb9FNnj4>

<https://stackoverflow.com/questions/22882074/download-image-and-its-associated-info-through-qr-code-on-android-app>

<https://stackoverflow.com/questions/45967649/firebase-authentication-error-in-android>

<https://firebase.google.com/docs/auth/android/google-signin>

<https://developer.android.com/studio/write/firebase.html>

<https://firebase.google.com/docs/database/android/start/>

Yours sincerely,

Ramya Radhakrishnakumar, Sushant Sharma & Vinokkumar Uthayakumar

Humber College

School of Applied Tech

## 7.2 Report 2

Sushant Sharma < [sushantsharma88888@gmail.com](mailto:sushantsharma88888@gmail.com) >

To: Austin Tian < [Austin.Tian@humber.ca](mailto:Austin.Tian@humber.ca) >

Cc: Ramya Radhakrishnakumar < [ramyarkkumar27@gmail.com](mailto:ramyarkkumar27@gmail.com) >, Vinokkumar Uthayakumar<[ukvino3@gmail.com](mailto:ukvino3@gmail.com) >

Dear Austin Tian,

This email is an overview of our progress during week 6 updated by student B.

This week we worked on the application and database connectivity with the hardware as well as the app. Sushant and Vino worked on validating the login, registering and authentication with the firebase database. Ramya spent some time researching on how to connect the firebase database with our raspberry pi, so that we can store our sensor’s scanned data into it.

We created a separate database for Merchant. We are able to store the value scanned by QR code scanner and add it to the merchandiser database.

Some of the problems/challenges encountered this week was, we tried to hide the navigation bar after the user logs in so that they stay logged in even if they click on different fragment/activity. We are also looking for a way to hide the login fragment when the user is already logged in until they click logout button.  We also imported the libraries for python to connect it to the database. We are up to date according to the schedule.

The financial status is the same as before. Most likely it will not change as we have acquired all the necessary equipment.

Links to the Media added on GitHub (<https://github.com/RamyaRadhakrishnakumar/ceng355)>:

<https://github.com/RamyaRadhakrishnakumar/ceng355/blob/master/week6/transfer.PNG>

<https://github.com/RamyaRadhakrishnakumar/ceng355/blob/master/week6/enrol.PNG>

<https://github.com/RamyaRadhakrishnakumar/ceng355/blob/master/week6/login.PNG>

Some of the websites used as reference for the application and database development are as follows:

<https://pypi.python.org/pypi/python-firebase/1.2>

<https://www.raspberrypi.org/forums/viewtopic.php?t=183455>

<https://www.youtube.com/watch?v=BJfVoaifnzc>

<https://repl.it/repls/WorrisomeArcticBlock>

<http://python-textbok.readthedocs.io/en/1.0/Variables_and_Scope.html>

Yours sincerely,

Ramya Radhakrishnakumar, Sushant Sharma & Vinokkumar Uthayakumar

Humber College

School of Applied Tech

## 7.3 Report 3

Vinokkumar Uthayakumar<[ukvino3@gmail.com](mailto:ukvino3@gmail.com) >

To: Austin Tian < [Austin.Tian@humber.ca](mailto:Austin.Tian@humber.ca) >

Cc: Ramya Radhakrishnakumar < [ramyarkkumar27@gmail.com](mailto:ramyarkkumar27@gmail.com) >, Sushant Sharma <[sushantsharma88888@gmail.com](mailto:sushantsharma88888@gmail.com) >

Dear Austin Tian,

This email is an overview of the progress during week 8 and week 9 updated by student C.

In the last two weeks, work was focused on the Graphical User Interface(GUI) for the sensors. Now the GUI for QR code scanner and fingerprint programs has been created. There is a plan to use LCD display for the sensors. Also, the team is working on subtracting the money from the bank account of the user after they have made a transaction(Pay/Transfer). Also, work was done on python code for fingerprint sensor to store information in the database based on the user’s fingerprint and email address provided.

There have been updates made to the layout of the application.

The problems being faced in the last 2 weeks are as follows, firstly the user information from the fire-base database was not being displayed. Here the user information could be seen from the database when debug is used. Finally, the information is being displayed on the overview such as the account number and account balance of the current user. Secondly, the object class is not working with fragments. The problem is, the buttons could not be transferred/passes from one class to another. The team tried fixing the issue by incorporating instance to make it work but it wasn’t successful. We are not able to call a method from one fragment to another fragment.

We are on track with the project schedule.

There is a recent purchase that adds up to the financial update of the project. An LCD display has been purchased for CAD $55.

Links to the Media added to GitHub (<https://github.com/RamyaRadhakrishnakumar/ceng355)>:

<https://github.com/RamyaRadhakrishnakumar/ceng355/blob/master/qr%20code%20scanner%20gui.PNG>

<https://github.com/RamyaRadhakrishnakumar/ceng355/blob/master/fingerprint%20gui.PNG>

<https://github.com/RamyaRadhakrishnakumar/ceng355/tree/master/Layouts%20of%20the%20application>

# 8. Conclusions

The mobile application and hardware has been developed in order to reduce the usage of cash and credit card. This would be more reliable because there wont be any miscellenaous handle of cash as well as cards. User can perform transactions way more secured compared to the online transfers. The QR code cannot be used without the user’s consent which is the speciality of E-money.

There are a number of planned features; more security, payment at the parking lot, time stamp and many more options for user friendly and interactive user interface.

# 9. Recommendations

The hardware is still in the prototype phase. The current cost for the single hardware is only about $347. This is the price for a version without a database. Code can be modified according to individual needs. Also the application is totally customized according to the need of the project. The cost of the hardware can be brought down to $200 by purchasing the sensors from third party stores.

# 10. Technical References/Bibliography

A. (2016, September 07). Firebase Web App Tutorial - Basics - Introduction / Adding Data to Database [ Part 1 ]. Retrieved from https://www.youtube.com/watch?v=F6UWb9FNnj4

Download Image and it's associated info through QR code on Android App. (n.d.). Retrieved from https://stackoverflow.com/questions/22882074/download-image-and-its-associated-info-through-qr-code-on-android-app

Firebase Authentication error in android. (n.d.). Retrieved from <https://stackoverflow.com/questions/45967649/firebase-authentication-error-in-android>

Authenticate Using Google Sign-In on Android  |  Firebase. (n.d.). Retrieved from https://firebase.google.com/docs/auth/android/google-signin

Connect to Firebase. (2018, March 26). Retrieved from https://developer.android.com/studio/write/firebase.html

Set up Firebase Realtime Database for Android  |  Firebase. (n.d.). Retrieved from https://firebase.google.com/docs/database/android/start/

Python-firebase. (n.d.). Retrieved from https://pypi.python.org/pypi/python-firebase/1.2

(n.d.). Retrieved from https://www.raspberrypi.org/forums/viewtopic.php?t=183455

Firebase And Python Tutorial 7 : How save or post data to firebase application. (2016, February 22). Retrieved from https://www.youtube.com/watch?v=BJfVoaifnzc

Repl.it. (n.d.). WorrisomeArcticBlock. Retrieved from <https://repl.it/repls/WorrisomeArcticBlock>

Variables and scope¶. (n.d.). Retrieved from http://python-textbok.readthedocs.io/en/1.0/Variables\_and\_Scope.html

# 11. Appendices

**11.1 Python Code Bank Account**

import sys

import re

import random

import math

import time

from firebase import firebase

from Tkinter import \*

from random import \*

root = Tk()

root.title("QR GUI Store")

root.geometry("590x410")

firebase = firebase.FirebaseApplication('https://qrmoney-1aec0.firebaseio.com')

#result = firebase.put('Transaction','Account Number',{'From': '','Account Balance': ''})

code = []

amount = []

final = []

finalbalance = []

uid = []

fbalance = []

l = Label(root, text="Please scan the QR Code: ")

l.grid(row = 0, column = 0)

entry = Entry()

entry.grid(row = 0, column = 1)

try:

while True:

def get():

'''Go and get the text from the entry'''

txt.delete(0.0, 'end')

code = (entry.get())

#amount = code.split(",")

#print amount[0]

#sentense = ("You paid", amount[1])

amount = re.split(r'[,.]',code)

sentense = ("You paid", amount[2])

print ("From: ", amount[0])

print ("To: ", amount[1])

print ("Amount: ", "$" + amount[2])

txt.insert(0.0, sentense)

uid = firebase.get('https://qrmoney-1aec0.firebaseio.com', None)

for x in uid:

if (amount[0] == x):

print("ok")

AccountBalance = firebase.get("https://qrmoney-1aec0.firebaseio.com/"+amount[1],"/Account Balance")

fromamount = firebase.get("https://qrmoney-1aec0.firebaseio.com/"+amount[0],"/Account Balance")

print("From (before tranfer): ", fromamount)

print("To (before transfer): ", AccountBalance)

finalbalance = float(amount[2]) + float(AccountBalance)

if(finalbalance != None):

x = randint(1,100)

ts = time.time();

#for x in range (0, 1000):

firebase.put("https://qrmoney-1aec0.firebaseio.com/"+amount[1],"/Account Balance", float(finalbalance))

firebase.put("https://qrmoney-1aec0.firebaseio.com/"+amount[1],'/Transaction' +str(x),{'From': amount[0] ,'To': amount[1] ,'amount': amount[2], 'Time': ts})

#for y in range (0, 1000):

fbalance = float(fromamount) - float(amount[2])

firebase.put("https://qrmoney-1aec0.firebaseio.com/"+amount[0],'Transaction'+str(x),{'From': amount[0],'To': amount[1],'amount':amount[2], 'Time': ts})

print("Time: ", ts)

#result = firebase.put('Customer','AccountInfo',{'Account Number': '','Account Balance': ''})

firebase.put("https://qrmoney-1aec0.firebaseio.com/"+amount[0],"/Account Balance", float(fbalance)

print("Final Balance after transfer in 'TO' account", finalbalance)

print("Final Balance after transfer in 'FROM' account", fbalance)

Button(text="Deposit", command=get).grid(row = 2, column = 0)

txt = Text(root, width=45, height=8)

txt.grid(row = 4, column = 0, columnspan = 2)

root.mainloop()

except KeyboardInterrupt:

print ("\nExit")

**11.2 Python Code Merchant**

import sys

from firebase import firebase

from Tkinter import \*

import time

import math

import random

from random import \*

root = Tk()

root.title("QR GUI Store")

root.geometry("410x280")

root.configure(background='orange')

firebase = firebase.FirebaseApplication('https://qr-merchant.firebaseio.com')

AccountBalance = firebase.get('https://qr-merchant.firebaseio.com','OUR DATABASE/17150R81400/Account Balance')

print (AccountBalance)

code = []

amount = []

final = []

finalbalance = []

ts = time.time()

l = Label(root, text="Please scan the QR Code: ",background="orange")

l.grid(row = 0, column = 0)

txt1 = Text(root, width=27, height=1)

txt1.grid(row = 0, column = 1)

txt = Text(root, width=45, height=8)

txt.grid(row = 4, column = 0, columnspan = 2)

try:

while True:

code = raw\_input("Please scan the QR Code: ")

amount = code.split(",")

txt1.insert(0.0, amount)

sentense = ("You paid", "$"+amount[1])

txt.insert(0.0, sentense)

finalbalance = float(amount[1]) + float(AccountBalance)

print(finalbalance)

x = randint(1,1000)

#result = firebase.put('Customer','AccountInfo'+amount[0],{'Account Number': amount[0],'Account Balance': amount[1], 'Time': ts})

#result = firebase.put('Customer','AccountInfo'+amount[0])

result = firebase.put('Customer',str(x) + 'AccountInfo'+amount[0],{'Account Number': amount[0],'Account Balance': amount[1], 'Time': ts})

firebase.put('https://qr-merchant.firebaseio.com','OUR DATABASE/17150R81400/Account Balance', format(finalbalance, '.2f'))

firebase.put('https://qr-merchant.firebaseio.com','OUR DATABASE/17150R81400/Time', ts)

#firebase.put('https://qr-merchant.firebaseio.com','Customer/AccountInfo'+str(amount[0])'/Account Balance', float(amount[1]))

#firebase.put('https://qr-merchant.firebaseio.com','Customer/AccountInfo'+str(amount[0])'/Account Number', amount[0])

root.mainloop()

except KeyboardInterrupt:

print ("\nExit")

**11.3 Python Code Fingerprint (Enroll, Search and Delete)**

import time

import os

from os import path

import tempfile

import json

import hashlib

import urllib, StringIO

from random import \*

from pyfingerprint.pyfingerprint import PyFingerprint

from firebase import firebase

from Tkinter import \*

import numpy as np

from google.cloud import storage

import subprocess as sub

import cv2

root = Tk()

root.title("FingerPrint GUI")

root.geometry("710x400")

code = []

email = []

uid = []

n = [100]

ts = time.time();

txt = Text(root, width=30, height=15)

txt2 = Text(root, width=30, height=15)

txt3 = Text(root, width=30, height=15)

firebase = firebase.FirebaseApplication('https://qrmoney-1aec0.firebaseio.com/')

os.environ["GOOGLE\_APPLICATION\_CREDENTIALS"] = "/home/pi/Desktop/qrmoney-ebd1ca9fcc42.json"

client = storage.Client()

bucket = client.get\_bucket('qrmoney-1aec0.appspot.com')

try:

f = PyFingerprint('/dev/ttyUSB0', 57600, 0xFFFFFFFF, 0x00000000)

if ( f.verifyPassword() == False ):

raise ValueError('The given fingerprint sensor password is wrong!')

except Exception as e:

print('The fingerprint sensor could not be initialized!')

#cent1 = ('The fingerprint sensor could not be initialized!')

#txt2.insert(0.0, cent1)

print('Exception message: ' + str(e))

exit(1)

def search():

print('Waiting for finger...')

#############################################################

#Wait that finger is read

while ( f.readImage() == False ):

pass

f.convertImage(0x01)

result = f.searchTemplate()

position = result[0]

print(position)

if ( position == -1 ):

print('No match found!')

exit(0)

else:

print('Found template at position #' + str(position))

## OPTIONAL stuff

##

## Loads the found template to charbuffer 1

f.loadTemplate(position, 0x01)

#Searchs template

############################################################################################################

print("ok1")

#giraffeBlob = bucket.get\_blob("image1")

print("ok2")

#URL = giraffeBlob.public\_url

print("ok3")

#req = urllib.urlopen(URL)

print("ok4")

#arr = np.asarray(bytearray(req.read()),dtype=np.uint8)

print("ok5")

#img = cv2.imdecode(arr,-1)

print("ok6")

#giraffeBlob.download\_as\_string()

print("ok7")

#file = cStringIO.StringIO(urllib.urlopen(URL).read())

print("ok8")

#img = Image.open(file) #img is my image from storage and f is from user so convert now

print("ok5")

#print(img)

#cv2.imshow("image", img)

print("ok6")

############################################################################################################

imageDestination = tempfile.gettempdir() + '/fingerprint.bmp'

########################################################

uid = firebase.get("https://qrmoney-1aec0.firebaseio.com/", None)

print("ok")

code = (entry.get())

print("ok")

email = code.split('.')

print("ok")

for x in uid:

print("ok")

#print(x)

###############################################################

if (email[0] == x):

print("ok")

test = firebase.get("https://qrmoney-1aec0.firebaseio.com/"+email[0],"/fingerprintImage")

print(test)

if(position == test):

test2 = firebase.get("https://qrmoney-1aec0.firebaseio.com/"+email[0],"/CardNumber")

test3 = firebase.get("https://qrmoney-1aec0.firebaseio.com/"+email[0],"/Account Balance")

#if(test == imageDestination):

print(test)

print(test2)

print(test3)

#imageDestination = tempfile.gettempdir() + '/fingerprint.bmp'

print('Downloading image (this take a while)...')

#f.downloadImage(test)

print("storeed")

print('The image was saved to "' + test + '".')

print(f)

print(imageDestination)

else:

#cent5 = (test)

#txt2(0.0, cent5)

print("Finger print is not matching with our datbase.....")

#root.mainloop()

else:

print("Your email id is not matching with our databse Please try agian later time")

def enroll():

sent4 = ('Waiting for finger...')

#label4 = Label(root, text="Waiting for same finger again...").grid()

print('Waiting for finger...')

while (f.readImage() == False):

pass

print ("Remove finger...")

sent2 = ("Remove finger...")

txt.insert(END, sent2)

print('Waiting for same finger again...')

#Wait that finger is read again

while ( f.readImage() == False ):

pass

#########################################################333

f.convertImage(0x02)

##########################################################

f.createTemplate()

positionNumber = f.storeTemplate()

imageDestination = tempfile.gettempdir() + '/fingerprint.bmp'

f.downloadImage(imageDestination)

#print ('The image was saved to "' + imageDestination + '".')

############################################################################################

#for k in range 10:

x = randint(1,100)

imageBlob = bucket.blob("image" +str(x))

imageBlob.upload\_from\_filename(imageDestination)

#num +=1

print("ok")

############################################################################################

#loop through all the uid

code = (entry.get())

uid = firebase.get("https://qrmoney-1aec0.firebaseio.com/", None)

email = code.split('.')

for x in uid:

print(x)

# if my email and uid is matches

if (email[0] == x):

firebase.put("https://qrmoney-1aec0.firebaseio.com/"+email[0],'/fingerprintImage', positionNumber)

print('Finger enrolled successfully!')

#pass

else:

print("Your email id is not matching with our databse Please try agian later time")

def delete():

print('Currently used templates: ' + str(f.getTemplateCount()) +'/'+ str(f.getStorageCapacity()))

## Tries to delete the template of the finger

print('Waiting for finger...')

while (f.readImage() == False):

pass

print ("Remove finger...")

sent2 = ("Remove finger...")

txt.insert(END, sent2)

print('Waiting for same finger again...')

while ( f.readImage() == False ):

pass

code = (entry.get())

uid = firebase.get("https://qrmoney-1aec0.firebaseio.com/", None)

email = code.split('.')

for x in uid:

print(x)

# if my email and uid is matche

if (email[0] == x):

positionNumber = firebase.get("https://qrmoney-1aec0.firebaseio.com/"+email[0],"/fingerprintImage")

#print('Finger enrolled successfully!')

#pass

else:

print("Your email id is not matching with our databse Please try agian later time")

#positionNumber = input('Please enter the template position you want to delete: ')

positionNumber = int(positionNumber)

if ( f.deleteTemplate(positionNumber) == True ):

print('Fingerprint has been deleted from our database!!!')

firebase.delete("https://qrmoney-1aec0.firebaseio.com/"+email[0],"/fingerprintImage")

#except Exception as e:

# print('Operation failed!')

#print('Exception message: ' + str(e))

#exit(1)

try:

label = Label(root, text="Please enter your email: ")

label.grid(row = 0, column = 1)

entry = Entry()

entry.grid(row = 1, column = 1)

label3 = Label(root)

label3.grid(row=2, column = 0)

enroll1 = Button(text="Enroll", command=enroll).grid(row = 4, column = 0)

search = Button(text="Search", command=search).grid(row = 4, column = 1)

delete = Button(text="Delete", command=delete).grid(row = 4, column = 2)

#txt = Text(root, width=45, height=15)

txt.grid(row = 6, column = 0)

#txt2 = Text(root, width=45, height=15)

txt2.grid(row = 6, column = 1)

txt3.grid(row = 6, column = 2)

root.mainloop()

except Exception as e:

print('Operation failed!')

print('Exception message: ' + str(e))

exit(1)

**11.4 Android java code Contact.java**

**package** com.example.susha.qrmoney;  
**import** android.widget.EditText;  
*/\*\*  
 \* Created by susha on 2018-03-02.  
 \*/***public class** contact {  
 String **acbal**;  
 String **cardNumber**, email, **password**;  
  
 **public** contact(EditText cardNumber, EditText email, EditText password) {  
 }  
 **public** contact(String cardNo, String Password) {  
 **cardNumber** = cardNo;  
 **password** = Password;  
 *// email = finalEmail;* }  
  
 **public void** setAcbal(String acbal){  
  
 **this**.**acbal** = acbal;  
 }  
 **public** String getaccbal (){  
  
 **return this**.**acbal**;  
 }  
 **public void** setCardNumber(String cardNumber){  
 **this**.**cardNumber** = cardNumber;  
 }  
  
 **public** String getCardNumber(){  
 **return this**.**cardNumber**;  
 }  
  
 **public void** setEmail(String email){  
 **this**.email = email;  
 }  
 **public** String getEmail(){  
 **return this**.email;  
 }  
  
 **public void** setPassword(String password){  
 **this**.**password** = password;  
 }  
  
 **public** String getPassword(){  
 **return this**.**password**;  
 }  
}

**11.5 Android java code db.java**

**package** com.example.susha.qrmoney;  
*/\*\*  
 \* Created by vino on 2018-04-01.  
 \*/***public class** db {  
 String **accountnumber** = **null**;  
 String **accountbalance** = **null**;

**public** db(String accountbalance, String accountnumber){  
 **this**.**accountbalance** = accountbalance;  
 **this**.**accountnumber** = accountnumber;  
}  
**public void** setAccountnumber(String accountnumber){  
 **this**.**accountnumber** = accountnumber;  
 }  
  
 **public void** setAccountbalance(String accountbalance) {  
 **this**.**accountbalance** = accountbalance;  
 }  
 **public** String getAccountnumber(){  
 **return this**.**accountnumber**;  
  
 }  
 **public** String getAccountbalance(){  
  
 **return this**.**accountbalance**;  
 }  
}

**11.6 Android java code Default.java**

**package** com.example.susha.qrmoney;  
  
**import** android.annotation.SuppressLint;  
**import** android.app.Activity;  
**import** android.app.AlertDialog;  
**import** android.app.FragmentManager;  
**import** android.content.Context;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.support.annotation.NonNull;  
**import** android.support.annotation.Nullable;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentActivity;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.support.v7.widget.CardView;  
**import** android.text.InputFilter;  
**import** android.text.Layout;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.FrameLayout;  
  
**import** com.google.firebase.auth.FirebaseAuth;  
**import** com.google.firebase.auth.FirebaseUser;  
**import** com.google.firebase.database.DatabaseReference;  
**import** com.google.firebase.database.FirebaseDatabase;  
  
**import static** java.security.AccessController.*getContext*;  
  
**public class** Default **extends** AppCompatActivity **implements** View.OnClickListener {  
  
  
 **static** CardView *logincardview*;  
 **static** CardView *enrolcardview*;  
 **static** CardView *paycardview*;  
 **static** CardView *transfercardview*;  
 **static** CardView *parkcardview*;  
 **static** Default *activityA*;  
 **private** FirebaseAuth **mAuth**;  
 FirebaseUser **currentUser**;  
 View **v**;  
  
 **public** Default(CardView logincardview, CardView enrolcardview, CardView paycardview, CardView transfercardview, CardView parkcardview) {  
 **this**.*logincardview* = logincardview;  
 **this**.*enrolcardview* = enrolcardview;  
 **this**.*parkcardview* = parkcardview;  
 **this**.*paycardview* = paycardview;  
 **this**.*transfercardview* = transfercardview;  
 }  
  
 **public** Default() {}  
  
  
  
 @Override  
 **protected void** onCreate(**final** Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_default***);  
 *activityA* = **this**;  
 FirebaseDatabase database;  
 **final** DatabaseReference users;  
 **mAuth** = FirebaseAuth.*getInstance*();  
 **currentUser** = **mAuth**.getCurrentUser();  
 *logincardview* = (CardView) findViewById(R.id.***Logindefualt***);  
 *enrolcardview* = (CardView) findViewById(R.id.***enroldefault***);  
 *paycardview* = (CardView) findViewById(R.id.***paydefault***);  
 *transfercardview* = (CardView) findViewById(R.id.***transferdefault***);  
 *parkcardview* = (CardView) findViewById(R.id.***parkingdefault***);  
 **boolean** clicked = **false**;  
 **mAuth** = FirebaseAuth.*getInstance*();  
 **final** FirebaseUser currentUser = **mAuth**.getCurrentUser();  
 **final** MainActivity mainActivity = **new** MainActivity();  
 *logincardview*.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
  
 Intent i = **new** Intent(Default.**this**, MainActivity.**class**);  
 Default.**this**.startActivity(i);  
  
 }  
 });  
  
 *enrolcardview*.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
  
 *//Intent f = new Intent(Default.this, MainActivity.class);  
 //Default.this.startActivity(f);  
 //MainActivity activity = (MainActivity) getActivity();  
 //MainActivity mainActivity1 = new MainActivity();  
 // mainActivity.updateUI(currentUser);* **int** CONTENT\_VIEW\_ID = 10101010;  
 FrameLayout frame = **new** FrameLayout(Default.**this**);  
 *//Layout frame = new R.id.fragmentcontainer;* frame.setId(CONTENT\_VIEW\_ID);  
 setContentView(frame, **new** FrameLayout.LayoutParams(  
 FrameLayout.LayoutParams.***MATCH\_PARENT***, FrameLayout.LayoutParams.***MATCH\_PARENT***));  
 Fragment fragment = **null**;  
 fragment = **new** Enrolfragment();  
  
 **if** (fragment != **null**) {  
 android.support.v4.app.FragmentManager fragmentManager = getSupportFragmentManager();  
 android.support.v4.app.FragmentTransaction ft = fragmentManager.beginTransaction();  
 ft.replace(CONTENT\_VIEW\_ID, fragment);  
 ft.commit();  
 }  
 **else**{  
  
 **final** AlertDialog.Builder alert = **new** AlertDialog.Builder(Default.**this**);  
 alert.setMessage(**" you are currently loged in"**);  
 alert.setTitle(**"Pay Alert"**);  
  
 alert.setNegativeButton(**"Cancel"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** id) {  
 dialog.cancel();  
 }  
 });  
 alert.show();  
 }  
  
 }  
 });  
  
 *paycardview*.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 **if** (currentUser != **null**) {  
  
 **int** CONTENT\_VIEW\_ID = 1011111;  
 FrameLayout frame = **new** FrameLayout(Default.**this**);  
  
  
 frame.setId(CONTENT\_VIEW\_ID);  
 setContentView(frame, **new** FrameLayout.LayoutParams(  
 FrameLayout.LayoutParams.***MATCH\_PARENT***, FrameLayout.LayoutParams.***MATCH\_PARENT***));  
 Fragment fragment = **null**;  
 fragment = **new** pay();  
  
 **if** (fragment != **null**) {  
 android.support.v4.app.FragmentManager fragmentManager = getSupportFragmentManager();  
 android.support.v4.app.FragmentTransaction ft = fragmentManager.beginTransaction();  
 ft.replace(CONTENT\_VIEW\_ID, fragment);  
 ft.commit();  
 }  
  
 }  
 **else**{  
 **final** AlertDialog.Builder alert = **new** AlertDialog.Builder(Default.**this**);  
 alert.setMessage(**"In order use this function, You Must Login First "**);  
 alert.setTitle(**"Pay Alert"**);  
  
 alert.setNegativeButton(**"Cancel"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** id) {  
 dialog.cancel();  
 }  
 });  
 alert.show();  
  
 }  
 }  
  
 });  
  
 *transfercardview*.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 **if** (currentUser == **null**) {  
  
 **final** AlertDialog.Builder alert = **new** AlertDialog.Builder(Default.**this**);  
 alert.setMessage(**"In order use this function, You Must Login First "**);  
 alert.setTitle(**"Transfer"**);  
 alert.setNegativeButton(**"Cancel"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** id) {  
 dialog.cancel();  
 }  
 });  
 alert.show();  
  
 }**else**{  
 **int** CONTENT\_VIEW\_ID = 12020202;  
 FrameLayout frame = **new** FrameLayout(Default.**this**);  
  
  
 frame.setId(CONTENT\_VIEW\_ID);  
 setContentView(frame, **new** FrameLayout.LayoutParams(  
 FrameLayout.LayoutParams.***MATCH\_PARENT***, FrameLayout.LayoutParams.***MATCH\_PARENT***));  
 Fragment fragment = **null**;  
 fragment = **new** transfer();  
  
 **if** (fragment != **null**) {  
 android.support.v4.app.FragmentManager fragmentManager = getSupportFragmentManager();  
 android.support.v4.app.FragmentTransaction ft = fragmentManager.beginTransaction();  
 ft.replace(CONTENT\_VIEW\_ID, fragment);  
 ft.commit();  
 }  
 }  
  
 }  
 });  
 *parkcardview*.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 **final** AlertDialog.Builder alert = **new** AlertDialog.Builder(Default.**this**);  
 alert.setMessage(**"This Service is not available for use yet "**);  
 alert.setTitle(**"Pay Parking"**);  
 alert.setNegativeButton(**"Cancel"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** id) {  
 dialog.cancel();  
 }  
 });  
 alert.show();  
 }  
  
 });  
  
 }  
 @Override  
 **public void** onClick(View v) {  
  
 }}

**11.7 Android java code Email.java**

**package** com.example.susha.qrmoney;  
  
**import** android.app.ProgressDialog;  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.graphics.ImageFormat;  
**import** android.media.Image;  
**import** android.os.AsyncTask;  
**import** android.widget.ImageView;  
**import** android.widget.Toast;  
  
**import** java.io.File;  
**import** java.util.Properties;  
  
**import** javax.activation.DataHandler;  
**import** javax.activation.DataSource;  
**import** javax.activation.FileDataSource;  
**import** javax.mail.Authenticator;  
**import** javax.mail.BodyPart;  
**import** javax.mail.Message;  
**import** javax.mail.MessagingException;  
**import** javax.mail.Multipart;  
**import** javax.mail.PasswordAuthentication;  
**import** javax.mail.Session;  
**import** javax.mail.Transport;  
**import** javax.mail.internet.InternetAddress;  
**import** javax.mail.internet.MimeBodyPart;  
**import** javax.mail.internet.MimeMessage;  
**import** javax.mail.internet.MimeMultipart;  
  
**import static** android.R.attr.***bitmap***;  
**import static** android.R.id.***message***;  
*//import static com.example.vino.project.R.id.image;  
//import static com.example.vino.project.R.id.img;  
  
/\*\*  
 \* Created by vino on 2017-08-05.  
 \*/***public class** Email **extends** AsyncTask <Void, Void, Void> {  
  
 **private** Context **context**;  
 **private** Session **session**;  
 String **email**;  
 **private** ProgressDialog **progressDialog**;  
 *//MainActivity m = new MainActivity(this.)* MainActivity **main** = **new** MainActivity();  
 *//File myDirtest = new File();* **public** Email (Context context, String email){  
 **this**.**context** = context;  
 **this**.**email** = email;  
  
  
  
  
 }  
  
  
 @Override  
 **protected** Void doInBackground(Void... params) {  
  
 *//main.SaveImage(bitmap);  
 //File mydirtest = new File(String.valueOf(main.myDir));* Properties properties = **new** Properties();  
 properties.put(**"mail.smtp.host"**, **"smtp.gmail.com"**);  
 properties.put(**"mail.smtp.socketFactory.port"**, **"465"**);  
 properties.put(**"mail.smtp.socketFactory.class"**, **"javax.net.ssl.SSLSocketFactory"**);  
 properties.put(**"mail.smtp.port"**, **"465"**);  
 properties.put(**"mail.smtp.auth"**, **"true"**);  
  
  
 **session** = Session.*getDefaultInstance*(properties, **new** Authenticator() {  
  
 **protected** PasswordAuthentication getPasswordAuthentication() {  
  
 **return new** PasswordAuthentication(email\_auth.***EMAIL***, email\_auth.***PASSWORD***);  
  
  
 }  
  
 });  
  
 **try** {  
  
 *// Multipart multipart = new MimeMultipart();* MimeMessage mimeMessage = **new** MimeMessage(**session**);  
 mimeMessage.setFrom(**new** InternetAddress(email\_auth.***EMAIL***));  
 mimeMessage.addRecipient(Message.RecipientType.***TO***, **new** InternetAddress(**email**));  
 MimeBodyPart messageBodyPart = **new** MimeBodyPart();  
 Multipart multipart = **new** MimeMultipart();  
 messageBodyPart = **new** MimeBodyPart();  
 String file = **"/sdcard/Pictures/Image.png"**;  
 String fileName = **"Image.png"**;  
 DataSource source = **new** FileDataSource(file);  
 messageBodyPart.setDataHandler(**new** DataHandler(source));  
 messageBodyPart.setFileName(fileName);  
 multipart.addBodyPart(messageBodyPart);  
 mimeMessage.setContent(multipart);  
 mimeMessage.setSubject(**"Your eMoney is here, please use scanner to scan it."**);  
 Transport.*send*(mimeMessage);  
  
 } **catch** (MessagingException e) {  
  
 e.printStackTrace();  
 }  
 **catch** (Exception e){  
 e.printStackTrace();  
 }  
  
  
 **return null**;  
 }  
  
 @Override  
 **protected void** onPostExecute(Void aVoid) {  
 **super**.onPostExecute(aVoid);  
 **progressDialog**.dismiss();  
 Toast.*makeText*(**context**, **"Email Sent"**, Toast.***LENGTH\_LONG***).show();  
  
  
 }  
  
 @Override  
 **protected void** onPreExecute() {  
 **super**.onPreExecute();  
 **progressDialog** = ProgressDialog.*show*(**context**,**"Sending Email"**, **"Please Wait...."**, **false**,**false**);  
 }  
}

**11.8 Android java code email\_auth.java**

**package** com.example.susha.qrmoney;  
  
*/\*\*  
 \* Created by susha on 2018-03-04.  
 \*/***public class** email\_auth {  
  
 **public static final** String ***EMAIL*** = **"qr.donotreply@gmail.com"**;  
 **public static final** String ***PASSWORD*** = **"humber123"**;  
}

**11.9 Android java code Enrolfragment.java**

**package** com.example.susha.qrmoney;  
  
**import** android.app.AlertDialog;  
**import** android.content.Context;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.net.Uri;  
**import** android.os.Bundle;  
**import** android.support.annotation.FloatRange;  
**import** android.support.annotation.NonNull;  
**import** android.support.annotation.Nullable;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentManager;  
**import** android.support.v4.app.FragmentTransaction;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.text.TextUtils;  
**import** android.util.Log;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.Toast;  
  
**import** com.google.android.gms.tasks.OnCompleteListener;  
**import** com.google.android.gms.tasks.Task;  
**import** com.google.firebase.auth.AuthResult;  
**import** com.google.firebase.auth.FirebaseAuth;  
**import** com.google.firebase.auth.FirebaseUser;  
**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.ServerValue;  
**import** com.google.firebase.database.ValueEventListener;  
  
**import** java.text.DateFormat;  
**import** java.util.Date;  
**import** java.util.HashMap;  
**import** java.util.Map;  
**import** java.util.regex.Pattern;  
  
**import static** android.content.ContentValues.***TAG***;  
**import static** java.text.DateFormat.*getDateTimeInstance*;  
  
  
**public class** Enrolfragment **extends** Fragment {  
 EditText **cardNumber**, **Email**, **Password**, **Confirmpassword**;  
 Button **Enrol**;  
 Button **Login**;  
 **private** FirebaseAuth **mAuth**;  
 Fragment **fragment**;  
 Fragment **frag**;  
 String **emailfinal**, **passwordfinal**, **cardnumberfinal**;  
 **int accountbalancefinal**;  
 DatabaseReference **myRef**;  
 FirebaseDatabase **database**;  
 String **Emailstr**, **passwordstr**, **confirmpassstr**, **emailPatten**;  
 String **cardstr**;  
 Button **loginenrol**;  
  
  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container,  
 Bundle savedInstanceState) {  
 View view = inflater.inflate(R.layout.***fragment\_enrolfragment***, **null**);  
 **cardNumber** = (EditText) view.findViewById(R.id.***cardtext***);  
 **Email** = (EditText) view.findViewById(R.id.***emailtxt***);  
 **Password** = (EditText) view.findViewById(R.id.***passtxt***);  
 **Confirmpassword** = (EditText) view.findViewById(R.id.***conpasstxt***);  
 **confirmpassstr** = **Confirmpassword**.getText().toString().trim();  
 **mAuth** = FirebaseAuth.*getInstance*();  
 **emailPatten** = **"[a-zA-Z0-9.\_-]+@[a-z]+\\.+[a-z]+"**;  
 Button Enrol = (Button) view.findViewById(R.id.***enrolbtn***);  
 **loginenrol** = (Button) view.findViewById(R.id.***loginbtn***);  
 Button cancelenrol = (Button) view.findViewById(R.id.***cancelbtn***);  
 *//ref.child("yourNode").updateChildren(map);* Enrol.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 createAccount(**Email**.getText().toString(),  
 **Password**.getText().toString());  
  
  
 }  
 });  
 **loginenrol**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 logn();  
 }  
  
  
 });  
  
  
 cancelenrol.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Enrolfragment.**this**.onDetach();  
  
 }  
 });  
 **return** view;  
 }  
  
 *///Writing to database* **public void** writeNewUser(String cardstr, String emails, String password, **int** accountbalance) {  
 **int** balance = 0;  
 contact user = **new** contact(**cardNumber**, **Email**, **Password**);  
 cardstr = **cardNumber**.getText().toString().trim();  
 emails = **Email**.getText().toString().trim();  
 password = **Password**.getText().toString().trim();  
 **database** = FirebaseDatabase.*getInstance*();  
 String[] parts = emails.split(Pattern.*quote*(**"."**));  
 Map map = **new** HashMap();  
 FirebaseUser user1;  
 **myRef** = **database**.getReference(parts[0]);  
 DatabaseReference usersRef = **myRef**.child(**"Email"**);  
 DatabaseReference usersRef1 = **myRef**.child(**"CardNumber"**);  
 DatabaseReference usersRef2 = **myRef**.child(**"Password"**);  
 DatabaseReference usersRef3 = **myRef**.child(**"Account Balance"**);  
 DatabaseReference userRef4 = **myRef**.child(**"time"**);  
 usersRef.setValue(emails);  
 usersRef1.setValue(cardstr);  
 usersRef2.setValue(password);  
 usersRef3.setValue(balance);  
 userRef4.setValue(ServerValue.***TIMESTAMP***);  
  
 }  
  
 **public** Fragment logn() {  
 **fragment** = **new** Loginfragment();  
 **return fragment**;  
 }  
  
 */\*public static String getTimeDate(long timeStamp){  
 try{  
 DateFormat dateFormat = getDateTimeInstance();  
 Date netDate = (new Date(timeStamp));  
 return dateFormat.format(netDate);  
 } catch(Exception e) {  
 return "date";  
 }  
 }\*/* @Override  
 **public void** onViewCreated(View view, @Nullable Bundle savedInstanceState) {  
 **super**.onViewCreated(view, savedInstanceState);}  
  
  
  
 @Override  
 **public void** onStart() {  
 **super**.onStart();  
 *// Check if user is signed in (non-null) and update UI accordingly.  
 // FirebaseUser currentUser = mAuth.getCurrentUser();  
 //updateUI(currentUser);* }  
  
  
 **private void** createAccount(String email, String password) {  
 **if** (!validateForm()) {  
 **return**;  
 }  
 **mAuth**.createUserWithEmailAndPassword(email, password)  
 .addOnCompleteListener(Enrolfragment.**this**.getActivity(), **new** OnCompleteListener<AuthResult>() {  
 @Override  
 **public void** onComplete(@NonNull Task<AuthResult> task) {  
 **if** (task.isSuccessful()) {  
 writeNewUser(**emailfinal**, **passwordfinal**, **cardnumberfinal**, **accountbalancefinal**);  
  
 AlertDialog.Builder alert = **new** AlertDialog.Builder(getContext());  
 alert.setMessage(**"You have Registered successfully, Please Click either Login or Cancel "**);  
 alert.setTitle(**"Registration successful"**);  
 alert.setNeutralButton(**"go Back"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 **fragment** = **new** Loginfragment();  
  
  
 }  
  
 });  
  
 alert.show();  
  
 } **else** {  
 *// If sign in fails, display a message to the user.* Toast.*makeText*(Enrolfragment.**this**.getActivity(), **"Authentication failed."**,  
 Toast.***LENGTH\_SHORT***).show();  
 updateUI(**null**);  
  
 }  
 }  
 });  
 }  
  
 **private boolean** validateForm() {  
 **boolean** valid = **true**;  
  
 **Emailstr** = **Email**.getText().toString();  
 **passwordstr** = **Password**.getText().toString();  
 **cardstr** = **cardNumber**.getText().toString().trim();  
 **confirmpassstr** = **Confirmpassword**.getText().toString().trim();  
 String emailPatten = **"[a-zA-Z0-9.\_-]+@[a-z]+\\.+[a-z]+"**;  
  
 **if** (TextUtils.*isEmpty*(**Emailstr**)) {  
 **Email**.setError(**"Required."**);  
 valid = **false**;  
 } **else** {  
 **Email**.setError(**null**);  
 }  
 **if** (!**Emailstr**.matches(emailPatten)) {  
 **Email**.setError(**"Email is not vaild"**);  
 valid = **false**;  
 }**else** {  
 **Email**.setError(**null**);  
 }  
  
 **if** (**cardstr**.length() < 10){  
 **cardNumber**.setError(**"CardNumber isn't valid"**);  
 }**else** {  
 **cardNumber**.setError(**null**);  
 }  
  
 **if** (TextUtils.*isEmpty*(**passwordstr**)) {  
 **Password**.setError(**"Required."**);  
 valid = **false**;  
 } **else** {  
 **Password**.setError(**null**);  
 }  
  
 **if** (!**passwordstr**.equals(**confirmpassstr**)) {  
 **Password**.setError(**"Password do not match"**);  
 valid = **false**;  
 }**else** {  
 **Password**.setError(**null**);  
 }  
 **return** valid;  
 }  
  
  
 **private void** updateUI(FirebaseUser user) {  
 **if** (user != **null**) {  
  
 } **else** {  
 }  
 }  
}

**11.9 Android java code Loginfragment.java**

**package** com.example.susha.qrmoney;  
  
**import** android.app.AlertDialog;  
**import** android.content.Context;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.content.SharedPreferences;  
**import** android.os.Bundle;  
**import** com.example.susha.qrmoney.db;  
**import** android.preference.PreferenceManager;  
**import** android.support.annotation.NonNull;  
**import** android.support.design.widget.NavigationView;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentManager;  
**import** android.support.v4.app.FragmentTransaction;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.text.TextUtils;  
**import** android.util.AttributeSet;  
**import** android.util.Log;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.FrameLayout;  
**import** android.widget.ProgressBar;  
**import** android.widget.Switch;  
**import** com.google.firebase.auth.FirebaseAuth;  
**import** com.google.firebase.auth.FirebaseUser;  
  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
**import** com.google.android.gms.tasks.OnCompleteListener;  
**import** com.google.android.gms.tasks.Task;  
**import** com.google.firebase.auth.AuthResult;  
**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.example.susha.qrmoney.Model.User;  
**import** com.example.susha.qrmoney.Overview;  
  
  
**import** java.sql.Array;  
**import** java.util.ArrayList;  
**import** java.util.regex.Pattern;  
**import static** android.content.ContentValues.***TAG***;  
  
  
**public class** Loginfragment **extends** Fragment {  
  
 FirebaseDatabase **database**;  
 DatabaseReference **users**;  
 ArrayList<db> **vino**;  
 EditText **cardNumber**, **Email**, **editPassword**;  
 Button **Enrol**, **Login**;  
 **private** Switch **savepass**;  
 Fragment **fragment**;  
 NavigationView **navigationView**;  
 DataSnapshot **dataSnapshot**;  
 String **msg** = **"Sign in error"**;  
 ProgressBar **progressBar**;  
 String **cardNo**, **us**;  
 String **password**, **cardno**, **ps**;  
 *// int cardno;* String **customToken**;  
 Enrolfragment **enrolfragment** = **new** Enrolfragment();  
 **private** FirebaseAuth **mAuth**;  
 String **accountbal**;  
 String **accountno**;  
 TextView **displaycard**, **displaybal**;  
 **int acc**;  
 **private** String **mCustomToken**;  
 ArrayList<User> **val**;  
  
 SharedPreferences **prefs**;  
 SharedPreferences.Editor **editor**;  
  
  
 String **uid** = **"welcometohumber"**;  
  
**public** Loginfragment(){}  
  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container,  
 Bundle savedInstanceState) {  
 View view = inflater.inflate(R.layout.***fragment\_loginfragment***, container, **false**);  
 **database** = FirebaseDatabase.*getInstance*();  
 **editPassword** = (EditText) view.findViewById(R.id.***passtxt1***);  
 **Enrol** = (Button) view.findViewById(R.id.***enrolbtn1***);  
 **Login** = (Button) view.findViewById(R.id.***loginbtn1***);  
 **savepass** = (Switch) view.findViewById(R.id.***savepass***);  
 **Email** = (EditText) view.findViewById(R.id.***loginemail***);  
 **mAuth** = FirebaseAuth.*getInstance*();  
 *//displaybal = (TextView)view.findViewById(R.id.displaybal);  
 //displaycard = (TextView)view.findViewById(R.id.displaynumber);* **prefs** = getActivity().getSharedPreferences(**"pref"**, Context.***MODE\_PRIVATE***);  
 **editor** = **prefs**.edit();  
  
 **Login**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 signIn(  
 **Email**.getText().toString(),  
 **editPassword**.getText().toString()  
 );  
 }  
 });  
  
  
 **Enrol**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 **fragment** = **new** Enrolfragment();  
  
 **if** (**fragment** != **null**) {  
 android.support.v4.app.FragmentManager fragmentManager = getFragmentManager();  
 android.support.v4.app.FragmentTransaction ft = fragmentManager.beginTransaction();  
 ft.replace(R.id.***fragmentcontainer***, **fragment**);  
 ft.commit();  
 }  
  
 }  
 });  
 **return** view;  
 }  
  
 @Override  
 **public void** onStart() {  
  
 FirebaseUser currentUser = **mAuth**.getCurrentUser();  
  
 **super**.onStart();  
  
 }  
 **public void** updateUI(FirebaseUser user) {  
 *//FirebaseUser cu = mAuth.getCurrentUser();* **if** (user != **null**) {  
 Fragment fragment = **new** Overview();  
 android.support.v4.app.FragmentManager fragmentManager = getFragmentManager();  
 android.support.v4.app.FragmentTransaction ft = fragmentManager.beginTransaction();  
 ft.replace(R.id.***fragmentcontainer***, fragment);  
 ft.commit();  
 }  
 **else**{  
  
 }  
  
 }  
  
 **public boolean** validateForm() {  
 **boolean** valid = **true**;  
  
 *//String email = mEmailField.getText().toString();* String email = **Email**.getText().toString();  
 **if** (TextUtils.*isEmpty*(email)) {  
 **Email**.setError(**"Required."**);  
 valid = **false**;  
 } **else** {  
 **Email**.setError(**null**);  
 }  
  
 String password = **editPassword**.getText().toString();  
 **if** (TextUtils.*isEmpty*(password)) {  
 **editPassword**.setError(**"Required."**);  
 valid = **false**;  
 } **else** {  
 **editPassword**.setError(**null**);  
 }  
  
 **return** valid;  
 }  
  
  
 **public void** signIn(**final** String email, String password) {  
  
 *// Log.d(TAG, "signIn:" + email);* Toast.*makeText*(Loginfragment.**this**.getActivity(), **"sigIn"** + email,  
 Toast.***LENGTH\_SHORT***).show();  
 **if** (!validateForm()) {  
 **return**;  
 }  
  
 *// [START sign\_in\_with\_email]* **mAuth**.signInWithEmailAndPassword(email, password)  
 .addOnCompleteListener(Loginfragment.**this**.getActivity(), **new** OnCompleteListener<AuthResult>() {  
 @Override  
 **public void** onComplete(@NonNull Task<AuthResult> task) {  
 **if** (task.isSuccessful()) {  
 Toast.*makeText*(Loginfragment.**this**.getActivity(), **"Successfully Logged in!!"**,  
 Toast.***LENGTH\_SHORT***).show();  
 **final** FirebaseUser user = **mAuth**.getCurrentUser();  
 **if** (user != **null**) {  
 *//Overview overview = new Overview();  
 // Fragment f = new Loginfragment()  
 // fragment = new Overview();  
 //updateUI(user);  
  
 //mainActivity.fragment2  
 //android.support.v4.app.FragmentManager fragmentManager = getFragmentManager();  
 //android.support.v4.app.FragmentTransaction ft = fragmentManager.beginTransaction();  
 //ft.replace(R.id.fragmentcontainer, fragment);  
 //ft.commit();* Intent i = **new** Intent(Loginfragment.**this**.getActivity(), MainActivity.**class**);  
 Loginfragment.**this**.startActivity(i);  
 }  
 **else**{  
 *//* }  
  
 } **else** {  
 *// If sign in fails, display a message to the user.* Toast.*makeText*(Loginfragment.**this**.getActivity(), **"Authentication failed."**,  
 Toast.***LENGTH\_SHORT***).show();  
 updateUI(**null**);  
 }  
  
 *// [START\_EXCLUDE]* **if** (!task.isSuccessful()) {  
 *//mStatusTextView.setText(R.string.auth\_failed);* Toast.*makeText*(Loginfragment.**this**.getActivity(), **"Authentication failed."**,  
 Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 });  
 }  
  
 **public void** bringdata(String accountbalance, String accountnu) {  
 User user = **new** User(**cardno**, **accountbal**);  
 **users**.addValueEventListener(**new** ValueEventListener() {  
 @Override  
 **public void** onDataChange(DataSnapshot dataSnapshot) {  
 **cardno** = dataSnapshot.child(**"CardNumber"**).getValue().toString();  
 **accountbal** = dataSnapshot.child(**"Account Balanace"**).getValue().toString();  
 }  
 @Override  
 **public void** onCancelled(DatabaseError databaseError) {  
  
 }  
 });  
  
 }  
}

**11.10 Android java code MainActivity.java**

**package** com.example.susha.qrmoney;  
  
**import** android.app.AlertDialog;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.support.design.widget.NavigationView;  
**import** android.support.transition.Slide;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentManager;  
**import** android.support.v4.view.GravityCompat;  
**import** android.support.v4.widget.DrawerLayout;  
**import** android.support.v7.app.ActionBarDrawerToggle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.support.v7.widget.CardView;  
**import** android.support.v7.widget.Toolbar;  
**import** android.view.Gravity;  
**import** android.view.Menu;  
**import** android.view.MenuItem;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.TextView;  
  
**import** com.google.firebase.auth.FirebaseAuth;  
**import** com.google.firebase.auth.FirebaseUser;  
**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.regex.Pattern;  
  
**import static** com.example.susha.qrmoney.Default.*activityA*;  
**import static** com.example.susha.qrmoney.Default.*enrolcardview*;  
**import static** com.example.susha.qrmoney.Overview.*displaybal*;  
**import static** com.example.susha.qrmoney.Overview.*displaycard*;  
  
  
**public class** MainActivity **extends** AppCompatActivity  
  
 **implements** NavigationView.OnNavigationItemSelectedListener {  
 **private** FirebaseAuth **mAuth**;  
 FirebaseUser **currentUser**;  
 FirebaseDatabase **database**;  
 DatabaseReference **users**;  
 Fragment **fragment2** = **null**;  
 String **test**;  
 String **test2**;  
 String **test3**;  
 String **c**, **d**;  
 *//boolean click = false;* Fragment **fragment** = **null**;  
 Button **loginbtnmain**, **enrolmain**, **paybtnmain**, **transferbtnmain**;  
 Loginfragment **loginfragment** = **new** Loginfragment();  
 Default **aDefault** = **new** Default();  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.***toolbar***);  
 setSupportActionBar(toolbar);  
  
 **loginbtnmain** = (Button) findViewById(R.id.***Logindefualt***);  
 **enrolmain** = (Button) findViewById(R.id.***enroldefault***);  
 **paybtnmain** = (Button) findViewById(R.id.***paydefault***);  
 **transferbtnmain** = (Button) findViewById(R.id.***transferdefault***);  
  
 **final** 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.addDrawerListener(toggle);  
 toggle.syncState();  
  
 View v = **null**;  
 NavigationView navigationView = (NavigationView) findViewById(R.id.***nav\_view***);  
 navigationView.setNavigationItemSelectedListener(**this**);  
  
 **mAuth** = FirebaseAuth.*getInstance*();  
 **final** FirebaseUser currentUser = **mAuth**.getCurrentUser();  
  
 **if** (currentUser != **null**) {  
 **fragment** = **new** Overview();  
 DrawerLayout drawer1 = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 drawer1.closeDrawer(GravityCompat.***START***);  
 }  
 **else** {  
 **fragment** = **new** Loginfragment();  
 *//drawer.closeDrawer(GravityCompat.END);* }  
  
  
  
 **if** (**fragment** != **null**) {  
 android.support.v4.app.FragmentManager fragmentManager = getSupportFragmentManager();  
 android.support.v4.app.FragmentTransaction ft = fragmentManager.beginTransaction();  
 ft.replace(R.id.***fragmentcontainer***, **fragment**);  
 ft.commit();}  
 }  
  
  
  
 **public void** updateUI(FirebaseUser user) {  
 *//FirebaseUser cu = mAuth.getCurrentUser();  
 //if (user != null) {  
  
 // Fragment fragment = new Enrolfragment();  
  
 // android.support.v4.app.FragmentManager fragmentManager = getSupportFragmentManager();  
 // android.support.v4.app.FragmentTransaction ft = fragmentManager.beginTransaction();  
 //ft.replace(R.id.fragmentcontainer, fragment);  
 //ft.commit();}  
  
 // else{  
  
 // }* }  
  
 @Override  
 **protected void** onDestroy() {  
 FirebaseAuth.*getInstance*().signOut();  
 **super**.onDestroy();  
 }  
 @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** onOptionsItemSelected(MenuItem item) {  
 **int** id = item.getItemId();  
 *//noinspection SimplifiableIfStatement  
 // if (id == R.id.action\_settings) {  
 //return true;  
 //}* **return super**.onOptionsItemSelected(item);  
 }  
  
  
 **public void** takedata(){  
 **database** = FirebaseDatabase.*getInstance*();  
 **mAuth** = FirebaseAuth.*getInstance*();  
 FirebaseUser currentUser = **mAuth**.getCurrentUser();  
 **if** (currentUser != **null**) {  
 String id = currentUser.getEmail();  
 *//String test = id.getEmail();* String[] partsfinal = id.split(Pattern.*quote*(**"."**));  
 **users** = **database**.getReference(partsfinal[0]);  
 **users**.addValueEventListener(**new** ValueEventListener() {  
 @Override  
 **public void** onDataChange(DataSnapshot dataSnapshot) {  
 **test** = dataSnapshot.child(**"CardNumber"**).getValue().toString();  
 **test2** = dataSnapshot.child(**"Account Balance"**).getValue().toString();  
 **test3** = dataSnapshot.getKey().toString();  
 }  
  
  
 @Override  
 **public void** onCancelled(DatabaseError databaseError) {  
  
 }  
 });  
 }**else**{  
 *//* }  
 }  
  
 @SuppressWarnings(**"StatementWithEmptyBody"**)  
  
 @Override  
 **public boolean** onNavigationItemSelected(MenuItem item) {  
 *//String test = "vino";  
 // Handle navigation view item clicks here.* Fragment fragment1 = **new** Loginfragment();  
 FirebaseUser currentUser = **mAuth**.getCurrentUser();  
 android.support.v4.app.FragmentManager fm = getSupportFragmentManager();  
 Fragment fragment = **null**;  
 *//Fragment fragment2 = null;* **int** id = item.getItemId();  
  
 **if** (id == R.id.***dashboard***) {  
 fragment = **new** Overview();  
  
 } **else if** (id == R.id.***paydefault***) {  
 fragment = **new** pay();  
 } **else if** (id == R.id.***pay***) {  
 fragment = **new** pay();  
  
 } **else if** (id == R.id.***transfer***) {  
 fragment = **new** transfer();  
  
 } **else if** (id == R.id.***wallet***) {  
  
 } **else if** (id == R.id.***logoutbtn***){  
 **final** AlertDialog.Builder alert = **new** AlertDialog.Builder(MainActivity.**this**);  
 alert.setMessage(**"Do you want to logout? "**);  
 alert.setTitle(**"Logout"**);  
  
 alert.setNeutralButton(**"No"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** id) {  
 dialog.cancel();  
 }  
 });  
 alert.setNegativeButton(**"Yes"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 FirebaseAuth.*getInstance*().signOut();  
 Intent i = **new** Intent(MainActivity.**this**, Default.**class**);  
 MainActivity.**this**.startActivity(i);  
  
 }  
 });  
 alert.show();  
 }  
  
 **if** (fragment != **null**) {  
 android.support.v4.app.FragmentManager fragmentManager = getSupportFragmentManager();  
 android.support.v4.app.FragmentTransaction ft = fragmentManager.beginTransaction();  
 ft.replace(R.id.***fragmentcontainer***, fragment);  
 ft.commit();  
  
 }  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 drawer.closeDrawer(GravityCompat.***START***);  
 **return true**;  
 }  
}

**11.11 Android java code Overview.java**

**package** com.example.susha.qrmoney;  
  
**import** android.app.AlertDialog;  
**import** android.content.Context;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.net.Uri;  
**import** android.os.Bundle;  
**import** android.support.annotation.NonNull;  
**import** android.support.annotation.Nullable;  
**import** android.support.design.widget.NavigationView;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentManager;  
**import** android.support.v4.app.FragmentTransaction;  
**import** android.util.Log;  
**import** android.view.LayoutInflater;  
**import** android.view.Menu;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.Button;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** com.example.susha.qrmoney.Model.User;  
**import** com.google.firebase.auth.FirebaseAuth;  
**import** com.google.firebase.auth.FirebaseUser;  
**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.Query;  
**import** com.google.firebase.database.ValueEventListener;  
**import** com.example.susha.qrmoney.Loginfragment;  
  
**import** java.util.ArrayList;  
**import** java.util.List;  
**import** java.util.Map;  
**import** java.util.regex.Pattern;  
**import** com.example.susha.qrmoney.pay;  
  
**import static** android.content.ContentValues.***TAG***;  
  
  
**public class** Overview **extends** Fragment {  
  
Button **logoutbtn**, **cancelbtn**;  
 Fragment **fragment**;  
 NavigationView **navigationView**;  
 **static** TextView *displaycard*, *displaybal*;  
 FirebaseDatabase **database**;  
 FragmentManager **fm**;  
 Loginfragment **fragm**;  
 DatabaseReference **users**;  
 FirebaseAuth **mAuth**;  
 FirebaseUser **user**;  
 String **cardno**;  
 String **accountbal**;  
 Bundle **bundle**;  
 String **test**, **test2**, **c**,**a**;  
  
 **public** Overview(){}  
  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container,  
 Bundle savedInstanceState) {  
 View view = inflater.inflate(R.layout.***fragment\_overview***, container, **false**);  
 *//MainActivity activity = (MainActivity) getActivity();  
 // activity.takedata();  
 //displaybal.setText(activity.test2);  
 //displaycard.setText(activity.test);  
  
 //view.autofill();  
 //view.refreshDrawableState();* **logoutbtn** = (Button) view.findViewById(R.id.***logoutbtn***);  
 *displaybal* = (TextView)view.findViewById(R.id.***displaybal***);  
 *displaycard* = (TextView)view.findViewById(R.id.***displaynumber***);  
 **fm** = getFragmentManager();  
 **cancelbtn** = (Button) view.findViewById(R.id.***cancelbtnover***);  
 *// MainActivity activity = (MainActivity) getActivity();  
 // activity.takedata();  
 //displaybal.setText(activity.test2);  
 //displaycard.setText(activity.test);* **logoutbtn**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 **final** AlertDialog.Builder alert = **new** AlertDialog.Builder(Overview.**this**.getActivity());  
 alert.setMessage(**"Do you want to logout? "**);  
 alert.setTitle(**"Logout"**);  
  
 alert.setNeutralButton(**"No"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** id) {  
 dialog.cancel();  
 }  
 });  
  
 alert.setNegativeButton(**"Yes"**, **new** DialogInterface.OnClickListener() {  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 FirebaseAuth.*getInstance*().signOut();  
 Intent i = **new** Intent(Overview.**this**.getActivity(), Default.**class**);  
 Overview.**this**.startActivity(i);  
  
 }  
 });  
 alert.show();  
 }  
 });  
  
 **cancelbtn**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent in = **new** Intent(Overview.**this**.getActivity(), Default.**class**);  
 Overview.**this**.getActivity().startActivity(in);  
 }  
 });  
  
 **return** view;  
  
 }  
 @Override  
 **public void** onStart() {  
  
 MainActivity activity = (MainActivity) getActivity();  
 activity.takedata();  
 *displaybal*.setText(activity.**test2**);  
 *displaycard*.setText(activity.**test**);  
 **c** = *displaycard*.getText().toString();  
 **a** = *displaybal*.getText().toString();  
 **super**.onStart();  
}  
}

**11.12 Android java code pay.java**

**package** com.example.susha.qrmoney;  
  
**import** android.app.AlertDialog;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.graphics.Bitmap;  
**import** android.os.Bundle;  
**import** android.support.annotation.Nullable;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentManager;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.ImageView;  
**import** android.widget.Toast;  
  
**import** com.google.firebase.auth.FirebaseAuth;  
**import** com.google.firebase.auth.FirebaseUser;  
**import** com.google.firebase.database.DataSnapshot;  
**import** com.google.firebase.database.DatabaseError;  
**import** com.google.firebase.database.ValueEventListener;  
**import** com.google.zxing.BarcodeFormat;  
**import** com.google.zxing.MultiFormatWriter;  
**import** com.google.zxing.WriterException;  
**import** com.google.zxing.common.BitMatrix;  
**import** com.journeyapps.barcodescanner.BarcodeEncoder;  
**import** com.example.susha.qrmoney.Loginfragment;  
**import** com.example.susha.qrmoney.Overview;  
  
**import static** java.lang.Float.*parseFloat*;  
**import static** java.lang.Integer.*parseInt*;  
  
  
**public class** pay **extends** Fragment {  
 Button **back**, **paybutton**;  
 EditText **amount**, **accountnumber**;  
 ImageView **img**;  
 Fragment **fragment**;  
 FirebaseAuth **mAuth**;  
 String **balance1**;  
 String **number**;  
 Bundle **bundle**;  
 String **bal**;  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container,  
 Bundle savedInstanceState) {  
 View view = inflater.inflate(R.layout.***fragment\_pay***, **null**);  
 **back** = (Button) view.findViewById(R.id.***btn12***);  
 **paybutton** = (Button) view.findViewById(R.id.***paybtn***);  
 **amount** = (EditText) view.findViewById(R.id.***paytxt***);  
 *//accountnumber = (EditText) view.findViewById(R.id.accountnumberpay);* **img** = (ImageView)view.findViewById(R.id.***img***);  
 **mAuth** = FirebaseAuth.*getInstance*();  
  
 **back**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 **fragment** = **new** Overview();  
 android.support.v4.app.FragmentManager fragmentManager = getFragmentManager();  
 android.support.v4.app.FragmentTransaction ft = fragmentManager.beginTransaction();  
 ft.replace(R.id.***fragmentcontainer***, **fragment**);  
 ft.commit();}  
  
 });  
  
  
 **return** view;  
 }  
   
  
 @Override  
 **public void** onViewCreated(View view, @Nullable Bundle savedInstanceState) {  
 **super**.onViewCreated(view, savedInstanceState);  
 **final** String[] balance = **new** String[1];  
 **final** FirebaseUser currentUser = **mAuth**.getCurrentUser();  
  
 **if** (currentUser != **null**) {  
 **paybutton**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 *// Loginfragment loginfragment = new Loginfragment();  
 //String text2Qr = amount.getText().toString().trim();* String text2Qr = **amount**.getText().toString();  
 *// String pay = accountnumber.getText().toString();* MainActivity activity = (MainActivity) getActivity();  
 activity.takedata();  
  
 FragmentManager fm = getFragmentManager();  
  
 **if** (*parseFloat*(text2Qr) > *parseFloat*(activity.**test2**)) {  
 **final** AlertDialog.Builder alert = **new** AlertDialog.Builder(pay.**this**.getActivity());  
 alert.setMessage(**"You do not have an siginfcant amount in your account! "**);  
 alert.setTitle(**"Error"**);  
  
 alert.show();  
  
 }  
 **else** {  
 String combined = **null**;  
 combined = activity.**test** + **","** + text2Qr;  
  
  
 MultiFormatWriter multiFormatWriter = **new** MultiFormatWriter();  
  
 **try** {  
  
 BitMatrix bitMatrix = multiFormatWriter.encode(combined, BarcodeFormat.***QR\_CODE***, 300, 300);  
 BarcodeEncoder barcodeEncoder = **new** BarcodeEncoder();  
 *//Bitmap bitmap = barcodeEncoder.createBitmap(bitMatrix);* Bitmap finalBitmap = barcodeEncoder.createBitmap(bitMatrix);  
  
 **img**.setImageBitmap(finalBitmap);  
 } **catch** (WriterException e) {  
 e.printStackTrace();  
  
 }  
 }  
 **if** ((*parseFloat*(text2Qr) < *parseFloat*(activity.**test2**))){  
 Float newbalace = *parseFloat*(activity.**test2**) - *parseFloat*(text2Qr);  
 activity.**users**.child(**"Account Balance"**).setValue(newbalace);  
  
 }  
  
 }  
 });  
  
 }  
 }  
}

**11.13 Android java code slideActivity.java**

**package** com.example.susha.qrmoney;  
  
**import** android.support.annotation.NonNull;  
**import** android.support.annotation.Nullable;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.view.ViewPager;

**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
  
**public class** slideActivity **extends** AppCompatActivity {  
 **private** ViewPager **viewPa**;  
 **private** SlideAdapter **myadapter**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_slide***);  
 **viewPa** = findViewById(R.id.***viewpager***);  
 **myadapter** = **new** SlideAdapter(**this**);  
 **viewPa**.setAdapter(**myadapter**);}  
}

**11.4 Android java code SlideAdapter.java**

**package** com.example.susha.qrmoney;  
  
**import** android.content.ClipData;  
**import** android.content.Context;  
**import** android.content.Intent;  
**import** android.graphics.Color;  
**import** android.support.annotation.NonNull;  
**import** android.support.v4.app.Fragment;  
**import** android.support.v4.app.FragmentActivity;  
**import** android.support.v4.view.PagerAdapter;  
**import** android.support.v4.widget.DrawerLayout;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.Button;  
**import** android.widget.ImageView;  
**import** android.widget.LinearLayout;  
**import** android.widget.TextView;  
**import** com.example.susha.qrmoney.R;  
  
*/\*\*  
 \* Created by vino on 2018-03-21.  
 \*/***public class** SlideAdapter **extends** PagerAdapter{  
  
  
 Context **context**;  
 LayoutInflater **inflater**;  
 Button **btnstart**;  
  
  
 **public int**[] **lst\_image** = {  
 R.drawable.***image\_1***,  
 R.drawable.***image\_2***,  
 R.drawable.***image\_3***,  
 R.drawable.***image\_4*** };  
  
 **public** String[] **lst\_title** = {  
 **"BANKING"**,  
 **"PAY"**,  
 **"TRANSFER"**,  
 **"PAY PARKING"** };  
  
 **public** String[] **lst\_description** = {  
  
 **"Banking with QIK pay is simple, all you need is your phone. No cards, No cash"**,  
 **"With QiK Pay feature, you can make a payment without the use of card OR cash. Just generate the QR and let them scan it. Done, it's that easy"**,  
 **"ansfer money to bank the with Qr code. Eliminate the use of Cards. It's easy with QR QIK Pay"**,  
 **"In near Future, QIK pay will take over All the Parking Lots"** };  
  
 **public int**[] **lst\_backgroundcolor** = {  
 Color.*rgb*(55,55,55),  
 Color.*rgb*(239,85,85),  
 Color.*rgb*(110,40,89),  
 Color.*rgb*(1,188,212),};  
  
**public** SlideAdapter(Context context){  
 **this**.**context** = context;  
}  
 @Override  
 **public int** getCount() {  
 **return lst\_title**.**length**;  
 }  
  
 @Override  
 **public boolean** isViewFromObject(@NonNull View view, @NonNull Object object) {  
 **return** (view==(LinearLayout)object);}  
  
 @NonNull  
 @Override  
 **public** Object instantiateItem(@NonNull ViewGroup container, **int** position) {  
 **inflater** = (LayoutInflater) **context**.getSystemService(**context**.***LAYOUT\_INFLATER\_SERVICE***);  
 View view = **inflater**.inflate(R.layout.***slide***,container,**false**);  
 LinearLayout layoutslide = view.findViewById(R.id.***slidelinearlayout***);  
 ImageView imgslide = (ImageView) view.findViewById(R.id.***slideimg***);  
 TextView txtitle = (TextView) view.findViewById(R.id.***txttitle***);  
 TextView txtdescription = (TextView) view.findViewById(R.id.***txtdescription***);  
 **btnstart** = (Button) view.findViewById(R.id.***btnstart***);  
 **btnstart**.setVisibility(View.***INVISIBLE***);  
 **if** (position == 3) {  
 **btnstart**.setVisibility(View.***VISIBLE***);  
  
 **btnstart**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent in = **new** Intent(SlideAdapter.**this**.**context**, Default.**class**);  
 SlideAdapter.**this**.**context**.startActivity(in);  
  
 }  
 });  
 }  
  
 layoutslide.setBackgroundColor(**lst\_backgroundcolor**[position]);  
 imgslide.setImageResource(**lst\_image**[position]);  
 txtitle.setText(**lst\_title**[position]);  
 txtdescription.setText(**lst\_description**[position]);  
 container.addView(view);  
 **return** view;  
  
 }  
  
 @Override  
 **public void** destroyItem(@NonNull ViewGroup container, **int** position, @NonNull Object object) {  
 container.removeView((LinearLayout)object);}  
}

**11.14 Android java code transfer.java**

**package** com.example.susha.qrmoney;  
  
**import** android.app.AlertDialog;  
**import** android.content.Context;  
**import** android.graphics.Bitmap;  
**import** android.net.Uri;  
**import** android.os.Bundle;  
**import** android.os.Environment;  
**import** android.support.annotation.Nullable;  
**import** android.support.v4.app.Fragment;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.ImageView;  
  
**import** com.google.zxing.BarcodeFormat;  
**import** com.google.zxing.MultiFormatWriter;  
**import** com.google.zxing.WriterException;  
**import** com.google.zxing.common.BitMatrix;  
**import** com.journeyapps.barcodescanner.BarcodeEncoder;  
  
**import** java.io.File;  
**import** java.io.FileOutputStream;  
**import** java.util.Random;  
**import** java.util.regex.Pattern;  
  
**import static** java.lang.Float.*parseFloat*;  
**import static** java.lang.Integer.*parseInt*;  
  
  
**public class** transfer **extends** Fragment {  
 Button **back**, **transfer**, **generate**;  
 EditText **to**, **subject**, **amount\_txt**, **who**;  
 ImageView **transferiamge**;  
 Fragment **fragment**;  
 ImageView **img**;  
 String **email** = **""**;  
 String **subject1** = **""**;  
 Bitmap **finalBitmap**, **map**;  
 String **root** = Environment.*getExternalStorageDirectory*().toString();  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container,  
 Bundle savedInstanceState) {  
 *// Inflate the layout for this fragment* View view = inflater.inflate(R.layout.***fragment\_transfer***, container, **false**);  
 **back** = (Button) view.findViewById(R.id.***btn13***);  
 **transfer** = (Button)view.findViewById(R.id.***transferbtn***);  
 **to** = (EditText) view.findViewById(R.id.***totxt***);  
 *//who = (EditText) view.findViewById(R.id.);  
 //who= (EditText) view.findViewById(R.id.subjecttxt);* **transferiamge** = (ImageView) view.findViewById(R.id.***transferimage***);  
 **generate** = (Button) view.findViewById(R.id.***gnbtn***);  
 **amount\_txt** = (EditText) view.findViewById(R.id.***mini***);  
  
 **return** view;  
 }  
  
  
 @Override  
 **public void** onViewCreated(View view, @Nullable Bundle savedInstanceState) {  
 **super**.onViewCreated(view, savedInstanceState);  
  
 **back**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
  
 **if** (view.getId() == R.id.***btn13***) {  
 **fragment** = **new** Overview();  
 }  
  
  
 **if** (**fragment** != **null**) {  
 android.support.v4.app.FragmentManager fragmentManager = getFragmentManager();  
 android.support.v4.app.FragmentTransaction ft = fragmentManager.beginTransaction();  
 ft.replace(R.id.***fragmentcontainer***, **fragment**);  
 ft.commit();  
  
 }  
 }  
 });  
  
  
 **generate**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 MainActivity activity = (MainActivity) getActivity();  
 activity.takedata();  
 String text2Qr = **amount\_txt**.getText().toString().trim();  
 String pay = **to**.getText().toString().trim();  
 MultiFormatWriter multiFormatWriter = **new** MultiFormatWriter();  
 **if** (*parseFloat*(text2Qr) > *parseFloat*(activity.**test2**)) {  
 **final** AlertDialog.Builder alert = **new** AlertDialog.Builder(transfer.**this**.getActivity());  
 alert.setMessage(**"You do not have an siginfcant amount in your account! "**);  
 alert.setTitle(**"Error"**);  
  
 alert.show();  
  
 }  
 **else** {  
 *//String combined = activity.test3 + "," + text2Qr;* String[] tofinal = pay.split(Pattern.*quote*(**"."**));  
 String combined = activity.**test3** + **","** + tofinal[0] + **","** + text2Qr;  
 **try** {  
 BitMatrix bitMatrix = multiFormatWriter.encode(combined, BarcodeFormat.***QR\_CODE***, 200, 200);  
 BarcodeEncoder barcodeEncoder = **new** BarcodeEncoder();  
 *//Bitmap bitmap = barcodeEncoder.createBitmap(bitMatrix);* **finalBitmap** = barcodeEncoder.createBitmap(bitMatrix);  
  
 **transferiamge**.setImageBitmap(**finalBitmap**);  
 SaveImage(**finalBitmap**);  
 *// show\_dialogqr(msgqr);* } **catch** (WriterException e) {  
 e.printStackTrace();  
  
 }  
 }  
  
 }  
 });  
  
 **transfer**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 sendEmail();  
 }  
 });  
  
  
  
 }  
  
 **public void** transaction(){  
  
 }  
 **public void** sendEmail(){  
  
 **email** = **to**.getText().toString().trim(); *// trim will remove spaces  
 // subject1 = subject.getText().toString().trim();  
 //Image Get* Email m = **new** Email(getActivity(),**email**); *// this is access the mail file from....* m.execute(); *// this is ecute alll file in asynce* }  
 **public void** SaveImage(Bitmap finalBitmap) {  
 **map** = finalBitmap;  
 ImageView imgtest = **transferiamge**;  
 imgtest.setImageBitmap(finalBitmap);  
 File myDir = **new** File(**root** + **"/Pictures"**);  
 myDir.mkdirs();  
 *// Random generator = new Random();  
 // int n = 10000;  
 // n = generator.nextInt(n);  
 //String fpath = "/SDCARD/Pictures/";  
 //String fname = "Image.png";* String fname = **"Image.png"**;  
 File file = **new** File(myDir, fname);  
 **if** (file.exists()) file.delete();  
  
 **try** {  
 FileOutputStream out = **new** FileOutputStream(file);  
 *// out.write(finalBitmap);* finalBitmap.compress(Bitmap.CompressFormat.***PNG***, 90, out);  
 out.flush();  
 out.close();  
  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 }  
  
  
 *//}* }  
}

**11.15 Android java code welcomeActivity.java**

**package** com.example.susha.qrmoney;  
  
**import** android.content.Intent;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.view.animation.Animation;  
**import** android.view.animation.AnimationUtils;  
**import** android.widget.Button;  
**import** android.widget.LinearLayout;  
  
**public class** welcomeActivity **extends** AppCompatActivity {  
 LinearLayout **l1**,**l2**;  
 Button **btnsub**;  
 Animation **uptodown**,**downtoup**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_welcome***);  
 **btnsub** = (Button)findViewById(R.id.***buttonsub***);  
 **l1** = (LinearLayout) findViewById(R.id.***l1***);  
 **l2** = (LinearLayout) findViewById(R.id.***l2***);  
 **uptodown** = AnimationUtils.*loadAnimation*(**this**,R.anim.***uptodown***);  
 **downtoup** = AnimationUtils.*loadAnimation*(**this**,R.anim.***downtoup***);  
 **l1**.setAnimation(**uptodown**);  
 **l2**.setAnimation(**downtoup**);  
  
 **btnsub**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent intent = **new** Intent(welcomeActivity.**this**, slideActivity.**class**);  
 welcomeActivity.**this**.startActivity(intent);  
 }  
 });  
 }  
 }

**11.16 Android java code ExampleInstrumentedTest.java**

**package** com.example.susha.qrmoney;  
  
**import** android.content.Context;  
**import** android.support.test.InstrumentationRegistry;  
**import** android.support.test.runner.AndroidJUnit4;  
  
**import** org.junit.Test;  
**import** org.junit.runner.RunWith;  
  
**import static** org.junit.Assert.\*;  
  
*/\*\*  
 \* Instrumented test, which will execute on an Android device.  
 \*  
 \** ***@see*** *<a href="http://d.android.com/tools/testing">Testing documentation</a>  
 \*/*@RunWith(AndroidJUnit4.**class**)  
**public class** ExampleInstrumentedTest {  
 @Test  
 **public void** useAppContext() **throws** Exception {  
 *// Context of the app under test.* Context appContext = InstrumentationRegistry.*getTargetContext*();  
  
 *assertEquals*(**"com.example.susha.qrmoney"**, appContext.getPackageName());  
 }  
}

**11.17 Android java code ExampleUnitTest.java**

**package** com.example.susha.qrmoney;  
  
**import** org.junit.Test;  
  
**import static** org.junit.Assert.\*;  
  
*/\*\*  
 \* Example local unit test, which will execute on the development machine (host).  
 \*  
 \** ***@see*** *<a href="http://d.android.com/tools/testing">Testing documentation</a>  
 \*/***public class** ExampleUnitTest {  
 @Test  
 **public void** addition\_isCorrect() **throws** Exception {  
 *assertEquals*(4, 2 + 2);  
 }  
}

**11.18 Android activity\_default.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:orientation="vertical"  
 android:padding="10dp"  
 android:background="#fcfcfc"  
 tools:context=".MainActivity"  
 android:gravity="center"  
 android:id="@+id/ll"**>  
  
  
  
 <**LinearLayout  
 android:clipToPadding="false"  
 android:gravity="center"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
  
 <**android.support.v7.widget.CardView  
 android:clickable="true"  
 android:onClick="onClick"  
 android:id="@+id/Logindefualt"  
 android:layout\_width="160dp"  
 android:layout\_height="190dp"  
 android:layout\_margin="10dp"  
 tools:ignore="OnClick"**>  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:background="@color/green"**>  
 <**ImageView  
 android:layout\_width="64dp"  
 android:layout\_height="64dp"  
 android:background="@drawable/login\_default"  
 android:padding="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textStyle="bold"  
 android:layout\_marginTop="10dp"  
 android:text="Login"** />  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:background="@android:color/black"  
 android:layout\_margin="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:text="Check your bank activities"  
 android:padding="5dp"  
 android:textColor="@android:color/black"**/>  
  
 </**LinearLayout**>  
 </**android.support.v7.widget.CardView**>  
 <**android.support.v7.widget.CardView  
 android:foreground="?android:attr/selectableItemBackground"  
 android:clickable="true"  
 android:layout\_width="160dp"  
 android:layout\_height="190dp"  
 android:layout\_margin="10dp"  
 android:id="@+id/enroldefault"**>  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:background="@color/yello"**>  
 <**ImageView  
 android:layout\_width="64dp"  
 android:layout\_height="64dp"  
 android:background="@drawable/signup\_defaultf"  
 android:padding="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textStyle="bold"  
 android:layout\_marginTop="10dp"  
 android:text="Enrol Now"**/>  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:background="@android:color/black"  
 android:layout\_margin="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:text="Enrol now have your bank on your hand"  
 android:padding="5dp"  
 android:textColor="@android:color/black"**/>  
  
 </**LinearLayout**>  
 </**android.support.v7.widget.CardView**>  
  
 </**LinearLayout**>  
 <**LinearLayout  
 android:clipToPadding="false"  
 android:gravity="center"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
 <**android.support.v7.widget.CardView  
 android:foreground="?android:attr/selectableItemBackground"  
 android:clickable="true"  
 android:layout\_width="160dp"  
 android:layout\_height="190dp"  
 android:layout\_margin="10dp"  
 android:id="@+id/paydefault"**>  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:background="@color/lightgray"**>  
 <**ImageView  
 android:layout\_width="64dp"  
 android:layout\_height="64dp"  
 android:background="@drawable/pay\_default3"  
 android:padding="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textStyle="bold"  
 android:layout\_marginTop="10dp"  
 android:text="Pay"**/>  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:background="@android:color/black"  
 android:layout\_margin="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:text="Pay your bills with our app"  
 android:padding="5dp"  
 android:textColor="@android:color/black"**/>  
  
 </**LinearLayout**>  
 </**android.support.v7.widget.CardView**>  
 <**android.support.v7.widget.CardView  
 android:foreground="?android:attr/selectableItemBackground"  
 android:clickable="true"  
 android:layout\_width="160dp"  
 android:layout\_height="190dp"  
 android:layout\_margin="10dp"  
 android:id="@+id/transferdefault"**>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:gravity="center"  
 android:orientation="vertical"  
 android:background="@color/colorlightcyan"**>  
  
 <**ImageView  
 android:layout\_width="64dp"  
 android:layout\_height="64dp"  
 android:background="@drawable/transfer\_default"  
 android:padding="10dp"**/>  
  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="10dp"  
 android:text="Transfer"  
 android:textStyle="bold"** />  
  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:layout\_margin="10dp"  
 android:background="@android:color/black"** />  
  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:padding="5dp"  
 android:text="Transfer money to your favourite one"  
 android:textColor="@android:color/black"** />  
  
 </**LinearLayout**>  
 </**android.support.v7.widget.CardView**>  
  
 </**LinearLayout**>  
 <**LinearLayout  
 android:clipToPadding="false"  
 android:gravity="center"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
 <**android.support.v7.widget.CardView  
 android:foreground="?android:attr/selectableItemBackground"  
 android:clickable="true"  
 android:layout\_width="340dp"  
 android:layout\_height="150dp"  
 android:layout\_margin="10dp"  
 android:id="@+id/parkingdefault"**>  
 <**LinearLayout  
  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:background="@color/colororangeplusyellow"**>  
 <**ImageView  
 android:layout\_width="64dp"  
 android:layout\_height="64dp"  
 android:background="@drawable/parking\_default"  
 android:padding="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textStyle="bold"  
 android:layout\_marginTop="10dp"  
 android:text="Pay Park"**/>  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:background="@android:color/black"  
 android:layout\_margin="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:text="Pay your Parking with us"  
 android:padding="5dp"  
 android:textColor="@android:color/black"**/>  
  
 </**LinearLayout**>  
 </**android.support.v7.widget.CardView**>  
 </**LinearLayout**>  
</**LinearLayout**>

**11.19 Android activity\_main.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/drawer\_layout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fitsSystemWindows="true"  
 tools:openDrawer="start"**>  
  
 <**include  
 layout="@layout/app\_bar\_main"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**/>  
  
 <**android.support.design.widget.NavigationView  
 android:id="@+id/nav\_view"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_gravity="start"  
 android:fitsSystemWindows="true"  
 app:headerLayout="@layout/nav\_header\_main"  
 app:menu="@menu/activity\_main\_drawer"  
 android:background="@color/colorPrimary"**/>  
  
</**android.support.v4.widget.DrawerLayout**>

**11.20 Android activity\_slide.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 tools:context="com.example.susha.qrmoney.slideActivity"**>  
  
 <**android.support.v4.view.ViewPager  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:id="@+id/viewpager"**>  
 </**android.support.v4.view.ViewPager**>  
</**LinearLayout**>

**11.21 Android activity\_welcome.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".welcomeActivity"  
 android:orientation="vertical"  
 android:background="@drawable/galaxy3"**>  
 <**LinearLayout  
 android:id="@+id/l1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="400dp"  
 android:orientation="vertical"**>  
  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:layout\_marginTop="150dp"  
 android:text="QIKMONEY"  
 android:textColor="@color/colorPrimary"  
 android:textSize="60sp"  
 android:textStyle="bold"** />  
 <**TextView  
 android:textColor="@color/backgroundcolor"  
 android:layout\_width="398dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:text="WE MAKE MONEY HANDLING VERY SIMPLE"  
 android:textAlignment="center"  
 android:textSize="30sp"  
 android:layout\_marginTop="50dp"**/>  
 </**LinearLayout**>  
 <**LinearLayout  
 android:id="@+id/l2"  
 android:layout\_marginTop="50dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:background="@drawable/btnimg"**>  
 <**Button  
 android:id="@+id/buttonsub"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Welcome"  
 android:layout\_gravity="center"  
 android:background="@drawable/buttonstyle"** />  
  
 </**LinearLayout**>  
  
  
  
  
</**LinearLayout**>

**11.22 Android app\_bar\_main.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="com.example.susha.qrmoney.Default"  
 android:theme="@style/AppTheme"**>  
 <**android.support.design.widget.AppBarLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:theme="@style/AppTheme.AppBarOverlay"**>  
 <**android.support.v7.widget.Toolbar  
 android:id="@+id/toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="?attr/actionBarSize"  
 android:background="?attr/colorPrimary"  
 app:popupTheme="@style/AppTheme.PopupOverlay"** />  
  
 </**android.support.design.widget.AppBarLayout**>  
 <**include layout="@layout/content\_main"** />  
</**android.support.design.widget.CoordinatorLayout**>

**11.23 Android content\_main.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"  
 tools:showIn="@layout/app\_bar\_main"  
 android:id="@+id/fragmentcontainer"  
 android:orientation="horizontal"**>  
  
</**LinearLayout**>

**11.24 Android contenttwo.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:orientation="vertical"  
 android:padding="10dp"  
 android:background="#fcfcfc"  
 android:gravity="center"  
 android:id="@+id/fragmentcontainer1"  
 tools:context=".Default"**>  
  
</**LinearLayout**>

**11.25 Android fragment\_defaultfragment.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:padding="10dp"  
 android:background="#fcfcfc"  
 android:gravity="center"  
 android:id="@+id/ll"**>  
 <**LinearLayout  
 android:clipToPadding="false"  
 android:gravity="center"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
  
 <**android.support.v7.widget.CardView  
 android:clickable="true"  
 android:id="@+id/Logindefualt"  
 android:layout\_width="160dp"  
 android:layout\_height="190dp"  
 android:layout\_margin="10dp"**>  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:background="@color/green"**>  
 <**ImageView  
 android:layout\_width="64dp"  
 android:layout\_height="64dp"  
 android:background="@drawable/login\_default"  
 android:padding="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textStyle="bold"  
 android:layout\_marginTop="10dp"  
 android:text="Login"** />  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:background="@android:color/black"  
 android:layout\_margin="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:text="Check your bank activities"  
 android:padding="5dp"  
 android:textColor="@android:color/black"**/>  
  
 </**LinearLayout**>  
 </**android.support.v7.widget.CardView**>  
 <**android.support.v7.widget.CardView  
 android:foreground="?android:attr/selectableItemBackground"  
 android:clickable="true"  
 android:layout\_width="160dp"  
 android:layout\_height="190dp"  
 android:layout\_margin="10dp"  
 android:id="@+id/enroldefault"**>  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:background="@color/yello"**>  
 <**ImageView  
 android:layout\_width="64dp"  
 android:layout\_height="64dp"  
 android:background="@drawable/signup\_defaultf"  
 android:padding="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textStyle="bold"  
 android:layout\_marginTop="10dp"  
 android:text="Enrol Now"**/>  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:background="@android:color/black"  
 android:layout\_margin="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:text="Enrol now have your bank on your hand"  
 android:padding="5dp"  
 android:textColor="@android:color/black"**/>  
  
 </**LinearLayout**>  
 </**android.support.v7.widget.CardView**>  
  
 </**LinearLayout**>  
 <**LinearLayout  
 android:clipToPadding="false"  
 android:gravity="center"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
 <**android.support.v7.widget.CardView  
 android:foreground="?android:attr/selectableItemBackground"  
 android:clickable="true"  
 android:layout\_width="160dp"  
 android:layout\_height="190dp"  
 android:layout\_margin="10dp"  
 android:id="@+id/paydefault"**>  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:background="@color/lightgray"**>  
 <**ImageView  
 android:layout\_width="64dp"  
 android:layout\_height="64dp"  
 android:background="@drawable/pay\_default3"  
 android:padding="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textStyle="bold"  
 android:layout\_marginTop="10dp"  
 android:text="Pay"**/>  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:background="@android:color/black"  
 android:layout\_margin="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:text="Pay your bills with our app"  
 android:padding="5dp"  
 android:textColor="@android:color/black"**/>  
  
 </**LinearLayout**>  
 </**android.support.v7.widget.CardView**>  
 <**android.support.v7.widget.CardView  
 android:foreground="?android:attr/selectableItemBackground"  
 android:clickable="true"  
 android:layout\_width="160dp"  
 android:layout\_height="190dp"  
 android:layout\_margin="10dp"  
 android:id="@+id/transferdefault"**>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:gravity="center"  
 android:orientation="vertical"  
 android:background="@color/colorlightcyan"**>  
  
 <**ImageView  
 android:layout\_width="64dp"  
 android:layout\_height="64dp"  
 android:background="@drawable/transfer\_default"  
 android:padding="10dp"**/>  
  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="10dp"  
 android:text="Transfer"  
 android:textStyle="bold"** />  
  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:layout\_margin="10dp"  
 android:background="@android:color/black"** />  
  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:padding="5dp"  
 android:text="Transfer money to your favourite one"  
 android:textColor="@android:color/black"** />  
  
 </**LinearLayout**>  
 </**android.support.v7.widget.CardView**>  
  
 </**LinearLayout**>  
 <**LinearLayout  
 android:clipToPadding="false"  
 android:gravity="center"  
 android:orientation="horizontal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
 <**android.support.v7.widget.CardView  
 android:foreground="?android:attr/selectableItemBackground"  
 android:clickable="true"  
 android:layout\_width="340dp"  
 android:layout\_height="150dp"  
 android:layout\_margin="10dp"  
 android:id="@+id/parkingdefault"**>  
 <**LinearLayout  
  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:background="@color/colororangeplusyellow"**>  
 <**ImageView  
 android:layout\_width="64dp"  
 android:layout\_height="64dp"  
 android:background="@drawable/parking\_default"  
 android:padding="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textStyle="bold"  
 android:layout\_marginTop="10dp"  
 android:text="Pay Park"**/>  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:background="@android:color/black"  
 android:layout\_margin="10dp"**/>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:text="Pay your Parking with us"  
 android:padding="5dp"  
 android:textColor="@android:color/black"**/>  
  
 </**LinearLayout**>  
 </**android.support.v7.widget.CardView**>  
 </**LinearLayout**>  
</**LinearLayout**>

**11.26 Android frangment\_enrolfragment.xml**

<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="com.example.susha.qrmoney.Enrolfragment"  
 android:id="@+id/enrolfinal"**>  
  
 <**ImageView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="160dp"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_alignParentTop="true"  
 android:src="@drawable/overview\_header"  
 android:id="@+id/imageView4"** />  
  
 *<!--* ***TODO: Update blank fragment layout*** *-->* <**EditText  
 android:id="@+id/cardtext"  
 android:layout\_width="331dp"  
 android:layout\_height="49dp"  
 android:layout\_alignLeft="@+id/emailtxt"  
 android:layout\_alignStart="@+id/emailtxt"  
 android:layout\_below="@+id/emailtxt"  
 android:hint="Client card Number/Username"  
 android:inputType="number"** />  
  
 <**EditText  
 android:id="@+id/emailtxt"  
 android:layout\_width="332dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@+id/imageView4"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="25dp"  
 android:hint="Email"  
 android:inputType="textEmailAddress"** />  
  
 <**EditText  
 android:id="@+id/passtxt"  
 android:layout\_width="332dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignEnd="@+id/cardtext"  
 android:layout\_alignRight="@+id/cardtext"  
 android:layout\_below="@+id/cardtext"  
 android:hint="Password"  
 android:inputType="textPassword"** />  
  
 <**EditText  
 android:id="@+id/conpasstxt"  
 android:layout\_width="330dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignEnd="@+id/emailtxt"  
 android:layout\_alignRight="@+id/emailtxt"  
 android:layout\_below="@+id/passtxt"  
 android:hint="Confirm Password"  
 android:inputType="textPassword"** />  
  
  
  
 <**TextView  
 android:id="@+id/backtologintxt"  
 android:layout\_width="134dp"  
 android:layout\_height="34dp"  
 android:layout\_alignParentBottom="true"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginBottom="14dp"  
 android:text="Back to Login"  
 android:textSize="18sp"  
 android:textStyle="bold"  
 tools:layout\_editor\_absoluteX="146dp"  
 tools:layout\_editor\_absoluteY="447dp"** />  
  
 <**Button  
 android:id="@+id/enrolbtn"  
 android:layout\_width="100dp"  
 android:layout\_height="45dp"  
 android:layout\_alignBaseline="@+id/cancelbtn"  
 android:layout\_alignBottom="@+id/cancelbtn"  
 android:layout\_alignLeft="@+id/conpasstxt"  
 android:layout\_alignStart="@+id/conpasstxt"  
 android:background="@drawable/buttonshape"  
 android:text="Enrol"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
  
 <**Button  
 android:id="@+id/loginbtn"  
 android:layout\_width="100dp"  
 android:layout\_height="45dp"  
 android:layout\_alignBottom="@+id/backtologintxt"  
 android:layout\_alignEnd="@+id/cancelbtn"  
 android:layout\_alignRight="@+id/cancelbtn"  
 android:background="@drawable/buttonshape"  
 android:text="login"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
  
 <**Button  
 android:id="@+id/cancelbtn"  
 android:layout\_width="100dp"  
 android:layout\_height="45dp"  
 android:layout\_above="@+id/loginbtn"  
 android:layout\_marginBottom="15dp"  
 android:layout\_toEndOf="@+id/backtologintxt"  
 android:layout\_toRightOf="@+id/backtologintxt"  
 android:background="@drawable/buttonshape"  
 android:text="@string/cancel"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"**/>  
  
</**RelativeLayout**>

**11.27 Android fragment\_loginfragment.xml**

<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="com.example.susha.qrmoney.Loginfragment"  
 android:id="@+id/loginfrag"**>  
  
  
 <**ImageView  
 android:layout\_width="match\_parent"  
 android:layout\_height="160dp"  
 android:src="@drawable/overview\_header"  
 android:id="@+id/imageView2" android:contentDescription="@string/todo"** />  
  
 *<!--* ***TODO: Update blank fragment layout*** *-->* <**EditText  
 android:id="@+id/loginemail"  
 android:layout\_width="331dp"  
 android:layout\_height="49dp"  
 android:layout\_below="@+id/imageView2"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="22dp"  
 android:hint="@string/email"  
 android:inputType="textEmailAddress"  
 android:ems="10"**/>  
  
 <**EditText  
 android:id="@+id/passtxt1"  
 android:layout\_width="331dp"  
 android:layout\_height="49dp"  
 android:layout\_below="@+id/loginemail"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="34dp"  
 android:hint="@string/password"  
 android:inputType="textPassword"  
 android:ems="10"**/>  
  
  
 <**Switch  
 android:id="@+id/savepass"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignEnd="@+id/passtxt1"  
 android:layout\_alignRight="@+id/passtxt1"  
 android:layout\_below="@+id/passtxt1"  
 android:text="@string/remember\_me"** />  
  
  
 <**TextView  
 android:id="@+id/notclienttxt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignBottom="@+id/enrolbtn1"  
 android:layout\_alignEnd="@+id/loginbtn1"  
 android:layout\_alignRight="@+id/loginbtn1"  
 android:text="@string/not\_a\_client"** />  
  
  
 <**Button  
 android:id="@+id/loginbtn1"  
 android:layout\_width="100dp"  
 android:layout\_height="45dp"  
 android:layout\_alignLeft="@+id/passtxt1"  
 android:layout\_alignStart="@+id/passtxt1"  
 android:layout\_below="@+id/savepass"  
 android:background="@drawable/buttonshape"  
 android:layout\_marginTop="48dp"  
 android:text="@string/login"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
  
 <**Button  
 android:id="@+id/fingerprintbtn"  
 android:layout\_width="100dp"  
 android:layout\_height="45dp"  
 android:layout\_alignBaseline="@+id/loginbtn1"  
 android:layout\_alignBottom="@+id/loginbtn1"  
 android:layout\_alignEnd="@+id/savepass"  
 android:layout\_alignRight="@+id/savepass"  
 android:background="@drawable/buttonshape"  
 android:text="@string/fplogin"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
  
 <**Button  
 android:id="@+id/enrolbtn1"  
 android:layout\_width="100dp"  
 android:layout\_height="45dp"  
 android:layout\_below="@+id/loginbtn1"  
 android:layout\_centerHorizontal="true"  
 android:background="@drawable/buttonshape"  
 android:text="@string/enrol"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
</**RelativeLayout**>

**11.29 Android fragment\_overview.xml**

<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="com.example.susha.qrmoney.Overview"  
 android:id="@+id/overviewlayout"**>  
  
 *<!--* ***TODO: Update blank fragment layout*** *-->* <**ImageView  
 android:id="@+id/imageView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="160dp"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_alignParentTop="true"  
 android:src="@drawable/overview\_header"** />  
  
 <**TextView  
 android:id="@+id/cardnumber"  
 android:layout\_width="match\_parent"  
 android:layout\_height="40dp"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_below="@+id/imageView"  
 android:layout\_marginTop="19dp"  
 android:text="Account Number"  
 android:textColor="@android:color/black"  
 android:textSize="28sp"  
 android:textAlignment="center"  
 android:textAllCaps="true"  
 android:background="@color/colorlightcyan"  
 android:textStyle="bold"**/>  
  
 <**TextView  
 android:id="@+id/displaynumber"  
 android:layout\_width="match\_parent"  
 android:layout\_height="40dp"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_below="@+id/cardnumber"  
 android:layout\_marginTop="17dp"  
 android:textColor="@android:color/black"  
 android:textSize="28sp"  
 android:hint="account number "  
 android:textAlignment="center"  
 android:textStyle="bold"**/>  
  
 <**TextView  
 android:id="@+id/overtxt"  
 android:layout\_width="match\_parent"  
 android:layout\_height="40dp"  
 android:layout\_alignParentEnd="true"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentRight="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_below="@+id/displaynumber"  
 android:layout\_marginTop="20dp"  
 android:ems="10"  
 android:text="show Balance"  
 android:textColor="@android:color/black"  
 android:textSize="28sp"  
 android:textAlignment="center"  
 android:textAllCaps="true"  
 android:background="@color/colorlightcyan"  
 android:textStyle="bold"**/>  
  
 <**TextView  
 android:id="@+id/displaybal"  
 android:layout\_width="match\_parent"  
 android:layout\_height="40dp"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_below="@+id/overtxt"  
 android:layout\_marginTop="15dp"  
 android:hint="balance display"  
 android:textColor="@android:color/black"  
 android:textSize="28sp"  
 android:textAlignment="center"  
 android:textStyle="bold"**/>  
  
 <**Button  
 android:id="@+id/logoutbtn"  
 android:layout\_width="100dp"  
 android:layout\_height="45dp"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_below="@+id/displaybal"  
 android:layout\_marginLeft="29dp"  
 android:layout\_marginStart="29dp"  
 android:layout\_marginTop="21dp"  
 android:background="@drawable/buttonshape"  
 android:text="Logout"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
  
 <**Button  
 android:id="@+id/cancelbtnover"  
 android:layout\_width="100dp"  
 android:layout\_height="45dp"  
 android:layout\_alignBaseline="@+id/logoutbtn"  
 android:layout\_alignBottom="@+id/logoutbtn"  
 android:layout\_alignParentEnd="true"  
 android:layout\_alignParentRight="true"  
 android:layout\_marginEnd="31dp"  
 android:layout\_marginRight="31dp"  
 android:background="@drawable/buttonshape"  
 android:text="Back"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
</**RelativeLayout**>

**11.30 Android fragment\_pay.xml**

<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="com.example.susha.qrmoney.transfer"**>  
  
 <**com.aldoapps.autoformatedittext.AutoFormatEditText  
 android:id="@+id/paytxt"  
 android:layout\_width="300dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentTop="true"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="40dp"  
 android:hint="Enter the Amount here:"  
 android:maxLength="8"  
 app:isDecimal="true"** />  
  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="3dp"  
 android:layout\_below="@+id/paytxt"  
 android:id="@+id/viewid"**/>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="400dp"  
 android:layout\_below="@+id/viewid"  
 android:id="@+id/linearlayout"  
 android:orientation="horizontal"  
 android:layout\_centerInParent="true"**>  
  
  
 </**LinearLayout**>  
  
 <**Button  
 android:id="@+id/paybtn"  
 android:layout\_width="118dp"  
 android:layout\_height="45dp"  
 android:layout\_alignBaseline="@+id/btn12"  
 android:layout\_alignBottom="@+id/btn12"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_marginLeft="37dp"  
 android:layout\_marginStart="37dp"  
 android:background="@drawable/buttonshape"  
 android:text="Pay"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
  
  
 <**Button  
 android:id="@+id/btn12"  
 android:layout\_width="118dp"  
 android:layout\_height="45dp"  
 android:layout\_alignParentEnd="true"  
 android:layout\_alignParentRight="true"  
 android:layout\_below="@+id/linearlayout"  
 android:layout\_marginEnd="63dp"  
 android:layout\_marginRight="63dp"  
 android:layout\_marginTop="55dp"  
 android:background="@drawable/buttonshape"  
 android:text="Back"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
  
 <**ImageView  
 android:id="@+id/img"  
 android:layout\_width="match\_parent"  
 android:layout\_height="400dp"  
 android:layout\_alignLeft="@+id/linearlayout"  
 android:layout\_alignStart="@+id/linearlayout"  
 android:layout\_below="@+id/viewid"  
 android:background="@color/lightgray"** />  
</**RelativeLayout**>

**11.31 Android fragment\_transfer.xml**

<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="com.example.susha.qrmoney.transfer"**>  
  
 *<!--* ***TODO: Update blank fragment layout*** *-->* <**EditText  
 android:id="@+id/totxt"  
 android:layout\_marginTop="40dp"  
 android:layout\_width="331dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignLeft="@+id/mini"  
 android:layout\_alignParentTop="true"  
 android:layout\_alignStart="@+id/mini"  
 android:hint="To:"  
 android:inputType="textEmailAddress"** />  
  
  
  
 <**com.aldoapps.autoformatedittext.AutoFormatEditText  
 android:id="@+id/mini"  
 android:layout\_width="331dp"  
 android:layout\_marginTop="20dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@+id/totxt"  
 android:layout\_centerHorizontal="true"  
 android:hint="Enter the Amount here:"  
 android:maxLength="8"  
 app:isDecimal="true"** />  
  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="3dp"  
 android:layout\_below="@+id/mini"  
 android:id="@+id/view"**/>  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="300dp"  
 android:id="@+id/linearlayout1"  
 android:layout\_below="@+id/view"  
 android:orientation="horizontal"  
 android:layout\_centerInParent="true"  
 android:background="@color/lightgray"**>  
  
 <**ImageView  
 android:id="@+id/transferimage"  
 android:layout\_width="match\_parent"  
 android:layout\_height="300dp"** />  
  
 </**LinearLayout**>  
  
  
 <**Button  
 android:id="@+id/gnbtn"  
 android:layout\_width="100sp"  
 android:layout\_height="45dp"  
 android:layout\_alignParentBottom="true"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_marginBottom="48dp"  
 android:layout\_marginLeft="33dp"  
 android:layout\_marginStart="33dp"  
 android:background="@drawable/buttonshape"  
 android:text="Generate"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
  
 <**Button  
 android:id="@+id/btn13"  
 android:layout\_width="100dp"  
 android:layout\_height="45dp"  
 android:layout\_alignBaseline="@+id/transferbtn"  
 android:layout\_alignBottom="@+id/transferbtn"  
 android:layout\_alignParentEnd="true"  
 android:layout\_alignParentRight="true"  
 android:layout\_marginEnd="30dp"  
 android:layout\_marginRight="30dp"  
 android:background="@drawable/buttonshape"  
 android:text="Back"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
  
 <**Button  
 android:id="@+id/transferbtn"  
 android:layout\_width="100dp"  
 android:layout\_height="45dp"  
 android:layout\_alignBaseline="@+id/gnbtn"  
 android:layout\_alignBottom="@+id/gnbtn"  
 android:layout\_centerHorizontal="true"  
 android:background="@drawable/buttonshape"  
 android:text="Transfer"  
 android:textColor="#FFFFFF"  
 android:textSize="15sp"** />  
</**RelativeLayout**>

**11.32 Android nav\_header\_main.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="@dimen/nav\_header\_height"  
 android:background="@drawable/header"  
 android:gravity="bottom"  
 android:orientation="vertical"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"**>  
  
 <**ImageView  
 android:id="@+id/logoimage"  
 android:layout\_width="120dp"  
 android:layout\_height="120dp"  
 android:paddingTop="@dimen/nav\_header\_vertical\_spacing"  
 app:srcCompat="@drawable/emoney"** />  
  
  
  
 <**TextView  
 android:id="@+id/logotext"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="25dp"  
 android:text="QR Banking"  
 android:textColor="@color/colorBlue"**/>  
</**LinearLayout**>

**11.28 Android slide.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:gravity="center"  
 android:id="@+id/slidelinearlayout"**>  
  
  
 <**ImageView  
 android:layout\_width="200dp"  
 android:layout\_height="200dp"  
 android:id="@+id/slideimg"**/>  
  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Title Here"  
 android:textColor="#fff"  
 android:textSize="30dp"  
 android:textStyle="bold"  
 android:layout\_marginTop="40dp"  
 android:id="@+id/txttitle"** />  
  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textColor="#fff"  
 android:textSize="17dp"  
 android:layout\_marginTop="40dp"  
 android:textAlignment="center"  
 android:id="@+id/txtdescription"  
 android:text="Description Here"** />  
  
<**Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Start"  
 android:id="@+id/btnstart"  
 android:textSize="15dp"  
 android:layout\_marginTop="40dp"  
 android:background="@drawable/buttonshape"  
 android:textColor="#fff"**/>  
</**LinearLayout**>

**11.28 Android activity\_main\_drawer.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**menu xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 tools:showIn="navigation\_view"**>  
  
 <**group android:checkableBehavior="single"**>  
 <**item  
 android:id="@+id/dashboard"  
 android:icon="@drawable/overview"  
 android:title="DashBoard"** />  
 <**item  
 android:id="@+id/pay"  
 android:icon="@drawable/pay"  
 android:title="Pay"** />  
 <**item  
 android:id="@+id/transfer"  
 android:icon="@drawable/transfer"  
 android:title="Transfer"** />  
 <**item  
 android:id="@+id/wallet"  
 android:icon="@drawable/wallet"  
 android:title="Wallet"** />  
 <**item  
 android:id="@+id/logoutbtn"  
 android:icon="@drawable/wallet"  
 android:title="Logout"** />  
 </**group**>  
  
</**menu**>