A
PROJECT REPORT
ON

VIRTUAL WALLET

B.Tech (CE) Sem-VI

In fulfillment of all requirements for
Bachelor of Technology
In
Computer Engineering

Submitted by

Jigar Nandha (CE-075)
Meet Paija (CE-080)
Tarang Parikh (CE-083)

Under the Guidance of
Prof. Siddharth P. Shah

DEPARTMENT OF COMPUTER ENGINEERING
FACULTY OF TECHNOLOGY,
DHARMSINH DESAI UNIVERSITY
COLLEGE ROAD, NADIAD- 387001

DHARMSINH DESAI UNIVERSITY NADIAD-387001, GUJARAT

CERTIFICATE

This is to certify that the project carried out on the subject of System Design Practices entitled "Virtual Wallet" and recorded in this report is a work of

1)Jigar Nandha	**ROLL NO: CE-075**	**ID: 14CEUXS085**
2)Meet Paija	**ROLL NO: CE-080**	**ID: 14CEUOG023**
3)Tarang Parikh	**ROLL NO: CE-083**	**ID: 14MHUOS122**

Of Department Of Computer Engineering, semester VI. They were involved in Project Development during the academic year 2016 - 2017.

Prof. Siddharth P. Shah
(Project Guide)
Department of Computer Engineering
Faculty of Technology,
Dharmsinh Desai University, Nadiad.

Dr. C. K. Bhensdadia,
Head, Department of Computer Engineering,
Faculty of Technology,
Dharmsinh Desai University, Nadiad.

Abstract

When the people wants to reduce the manual money transaction at that time this type of wallets used.

Anyone who want to transfer a money or pay to someone if they have an account in this application then they can do it. For example, if the people wants to request to someone for money so using this chat facility they can use and also they get a notification if they have any update.

If the user wants to transfer money to another person then they also have an option which mode they want to transact money by QR code facility or manual mode.

Contents

1. Introduction

1.1 Project

Virtual Wallet is an Android application. When we want to exchange the money or pay by the e-wallet at that time we use this android application. Virtual Wallet helps you pay money through animations and it's abstract introduction. The animations help you understand every crucial concept of virtual wallet. It also allows you to send a message to another person that how much money you want to receive or send and also allows you to send using a QR code or receiving by this.

1.2 Technology Used

Front End Tool: Android Studio

Android Studio is the official Integrated Development Environment (IDE) for Android platform development. It is used to develop Android Applications. It internally uses Java programming language and XML.

Language: Java.

Java is a general-purpose computer programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation.

Markup Language: XML

In computing, **Extensible Markup Language** (**XML**) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

Diagram Tool: Star UML

StarUML is a UML tool by MKLab. It is used for creating UML diagrams and other various types of diagram.

2. Software Requirement Specifications

2.1 Purpose

The purpose of this document is to convey the functional and performance requirements of the product. This document ensures that the software requirements are properly understood. Also, the details of this document provide guidelines for future use and development of the product.

2.2 Scope

- User doesn’t perform actual transaction
- User authentication
- Reliable communication system
- Display the details of user account and all user transaction history
- User send money or receive money from other user and also request through messages
- “ACID” property of a transaction is successfully managed in our Application.

2.3 Definitions, Acronyms, and Abbreviations

Virtual Wallet or e-Wallet

2.4 Hardware Requirements

- Smartphones with android OS will be the target audience

2.5 Software Process Model

- The application would be developed using the Agile method of software development.
- Agile software development (Agile) is a collection of software development methodologies that promote adaptive planning,

evolutionary development and delivery, continuous improvement, and a time-boxed period of time to complete a body of work.

- This method is usually adopted when the project size is relatively small, the experience of the developers in the domain is less and the requirements aren’t clear as well as they are volatile.

Advantages of Agile model:

- New features are delivered quickly and frequently, with a high level of predictability.
- As the project is divided into manageable and small chunks the progress is visible and transparent.
- Changes can be easily adapted upon.

2.6 User Characteristics

- **End User**:

The user should have basic knowledge of the Android operating system.

- **Administrator**:

The administrator should have the knowledge about handling databases.

2.7 Nonfunctional Requirements

- Response time will be kept as low as possible in order to ensure promptness for the user.
- Usability: The user-friendly user interface must be designed in order to ensure ease of use.

2.8 Installation Requirements

The system to be developed doesn’t require any other software to be present on client’s machine.

2.9 Functional Requirements

R1: User Management

R.1.1 New user

Input: registration details

Outputs: confirmation message for registration

Processing: system will verify user details and register the user.

R.1.2 Manage user details

R.1.2.1 Display details

Input: select profile details

Output: system displays user information

Processing: system fetch current user details and display it.

R1.2.2 System provides a facility to change user information

Input: username or phone no

Output: confirmation message for updating details.

Processing: System verifies values and updated to the customer database.

R2: Login facility to authenticate valid user

Input: username and password

Output: welcome screen for valid user

Processing: system verifies username and password and proceed further

R3: System Provides Transfer and Request Money Facility

R3.1 System provides a facility to transfer money to other user accounts

Input: receiver's mobile number, amount.

Output: status message

Processing: System verifies details, transfer the money to receiver account

and updated the current balance in both accounts.

R3.2 System provide request the money facility

R3.2.1 System provides a facility to send the money request

Input: receiver's mobile number, amount.

Output: status message

Processing: System verify details, sent the request to receiver.

R3.2.2 System provides a facility to accept the request

Input: alert box asking for send the money or not

Output: confirmation message for request done

Processing: System checks current balance and then transfer

money to requested account

R4: System Provides facility to transfer money through QR Code

R4.1: System provides a facility to generate the QR Code

Input: generate QR Code flag

Output: QR code

Processing: system generates QR by using current user's mobile no.

R4.2 System provide facility to scan QR code and transfer particular amount

Input: QR code, amount

Output: confirmation message for transfer done

Processing: system scan particular QR code, verify input amount and transfer it.

R5: System Provide facility to chat with another user

Input: selected user, chat message

Output: chat screen

Processing: system check selected user and the message to that user.

R6: System provide facility to show all transaction history details

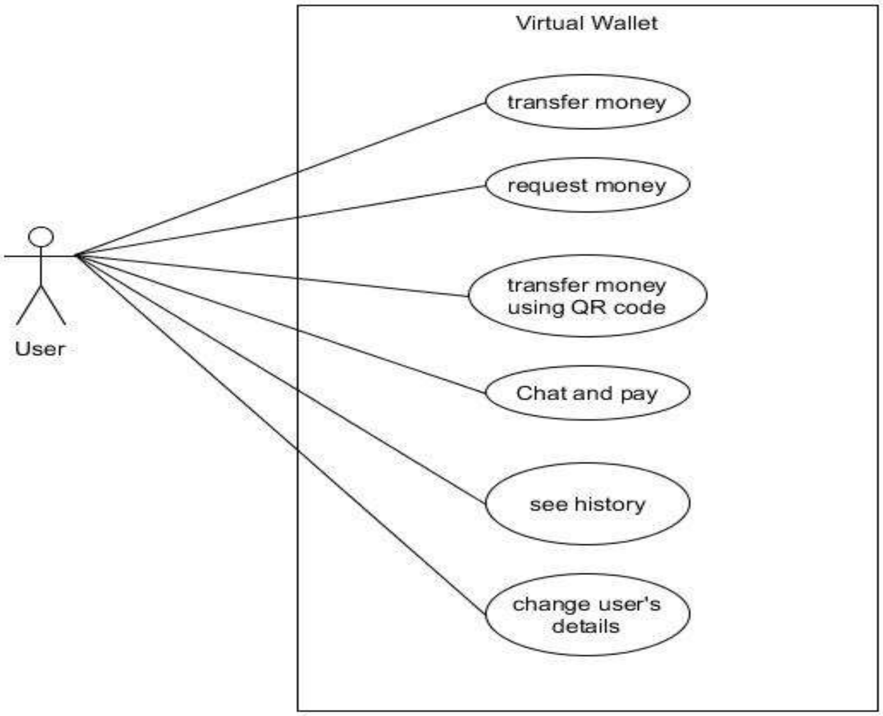
Input: history details flag

Output: all transaction history

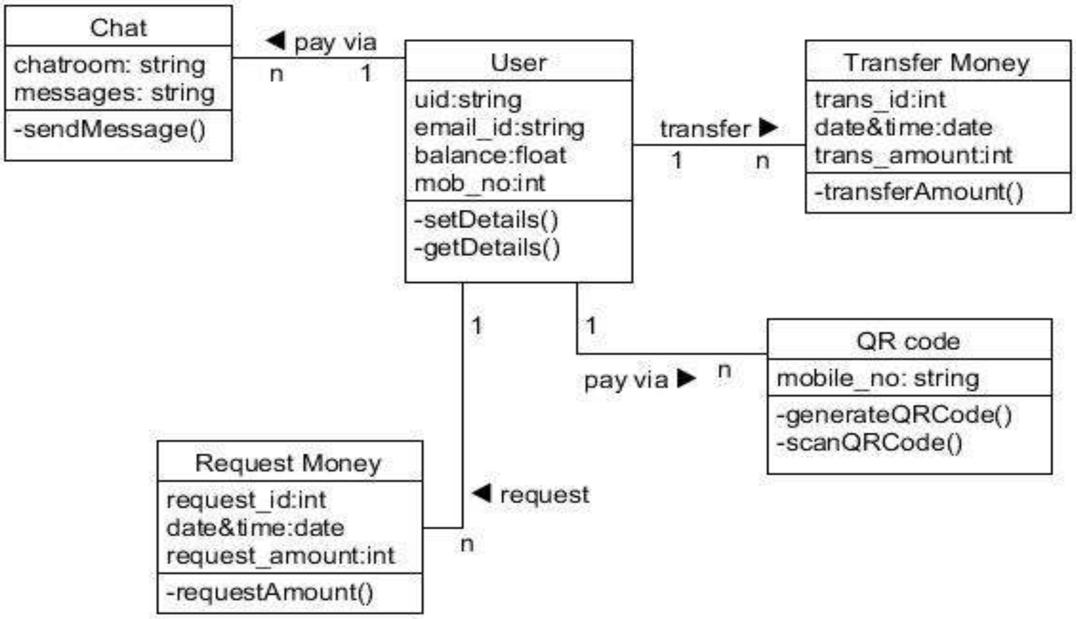
Processing: system fetch all history details from current user’s database and show that details.

3. System Design

3.1 Use-case Diagram

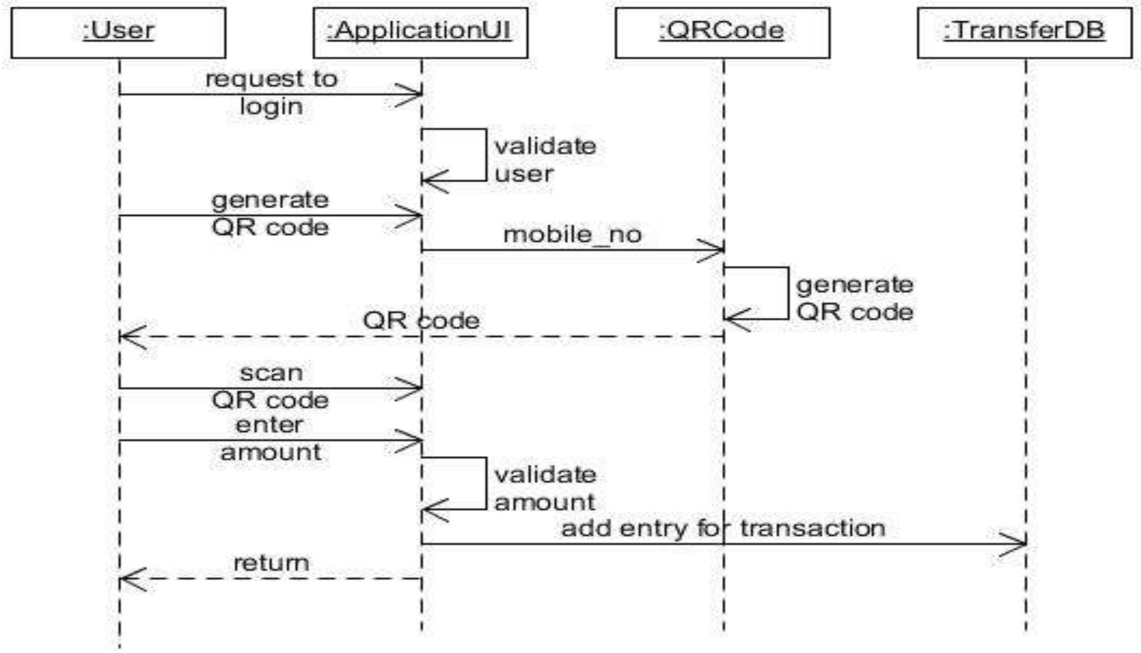


3.2 Class Diagram



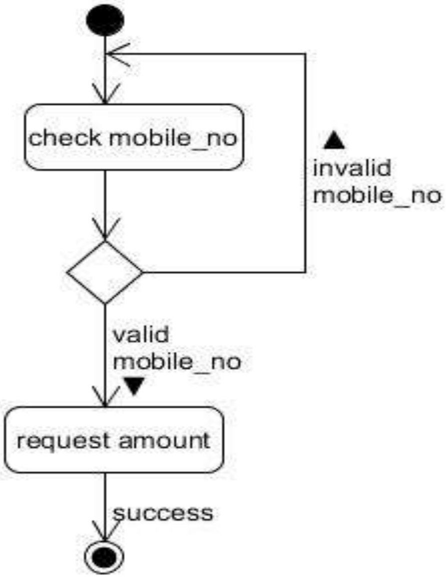
3.3 Sequence Diagram

Transfer money via QR Code

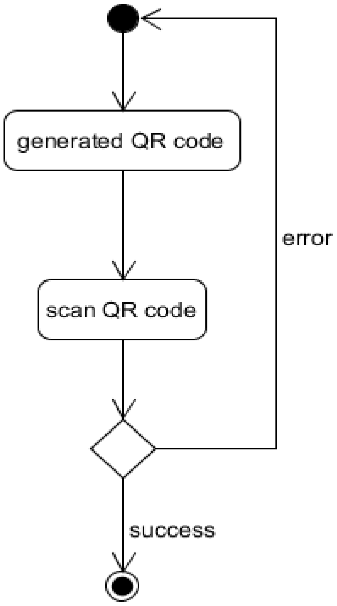


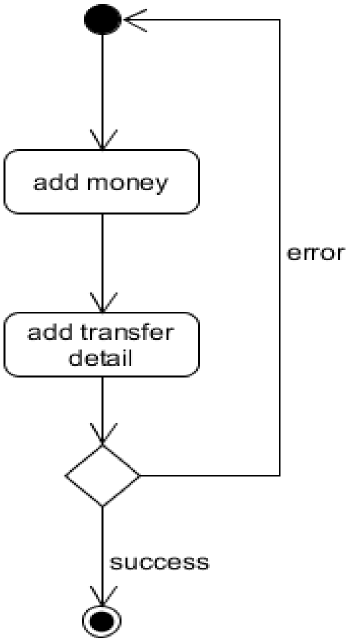
3.4 Activity Diagrams

3.4.1 Request Money



3.4.2 Transfer money





4. Implementation

4.1 Implementation Environment

- Android Studio
- XML
- Java

4.2 Modules Description

- **Abstract Introduction**:

  Abstract**:** It is used for transfer money
  Input parameter**:** money details
  Output parameter**:** respective update of database
  Processing logic**:** The introduction for the transfer money will be displayed in the web view.

5. Testing

5.1 Testing Plan

The testing is a technique that is going to be used in the project is black box testing the expected inputs to the system are applied and only the outputs are checked.

5.2 Testing Strategy

The development process repeats this testing sub-process a number of lines for the following phases.

- Unit Testing
- Integration Testing

Unit Testing tests a unit of code after coding of that unit is completed. Integration Testing tests whether the previous programs that make up a system, interface with each other as desired. System testing ensures that the system meets its stated design specifications. Acceptance testing is testing by users to ascertain whether the system developed is a correct implementation of the software requirements specification. Testing is carried out in such a hierarchical manner to that each component is correct and the assembly/combination of the component is correct. Merely testing a whole system at the end would most likely throw up errors in a component that would be very costly to trace and fix. We have performed both Unit Testing and System Testing to detect and fix errors.

5.3 Testing Methods

We have performed Black-box testing for the testing purpose. A brief description is given below:

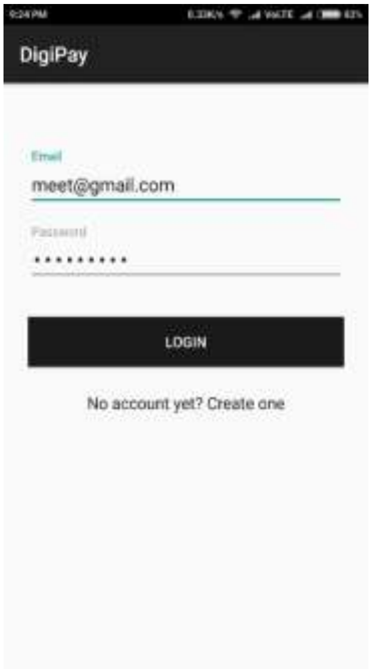
Black-box testing is a method of software testing that examines the functionality of an application without peering into its internal structures or workings. This method of test can be applied to virtually every level of software testing: unit, integration, system and acceptance. It typically comprises most if not all higher level testing, but can also dominate unit testing as well.

5.4 Test Cases

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results
T01	Authenticated user	1.enter email id 2. enter password	Emil id and password	Status message and Homepage	As per expected
T02	Transfer money	1.select option 2.generated QR code 3.scan QR code	QRcode	Prompt for money And details	As per expected
T03	Request for money	1.enter amount	Correctly entered amount	Correctly send	As per expected
T04	Update profile	1.input personal detail	Correctly enter details	Correctly updated status shown	As per expected
T05	View account	1.select option	Correctly Fetch from database	Account details	As per expected
T06	Chat with Other User	1.select user 2.send message	Correctly Entered Message	Chat Screen With messages	As per expected

6. Screenshots

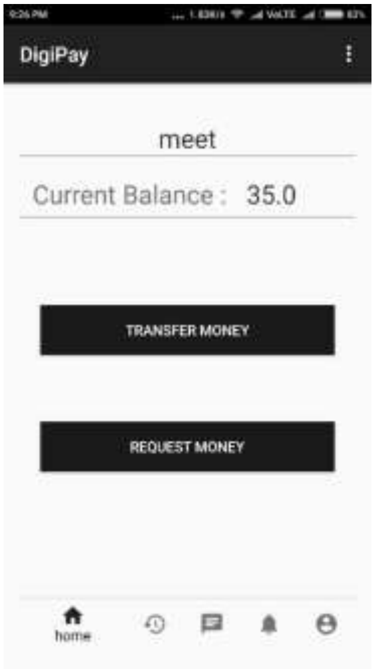
1.login	**2. Welcome Screen**



Email
meet@gmail.com

Password
•••••••••

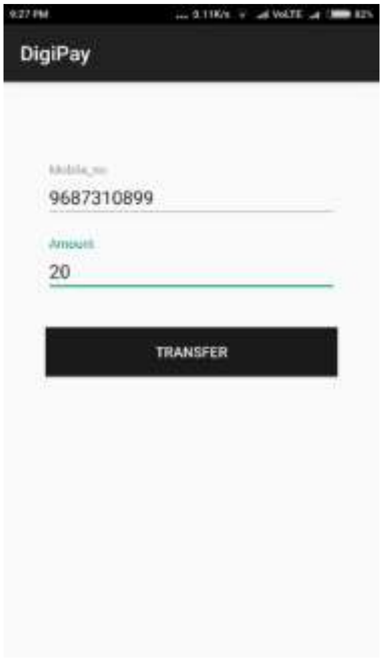
No account yet? Create one



meet

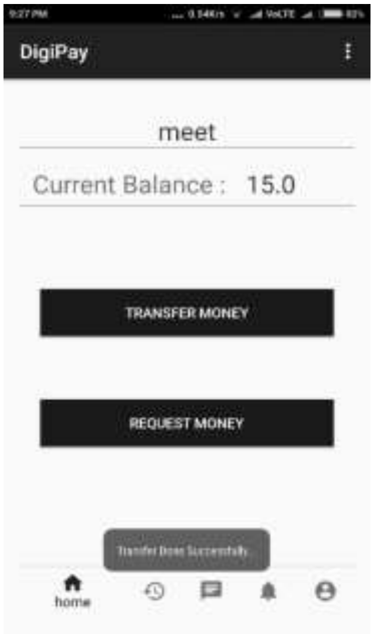
Current Balance : 35.0

3. Transfer Screen	**4.**



Mobile_no
9687310899

Amount
20

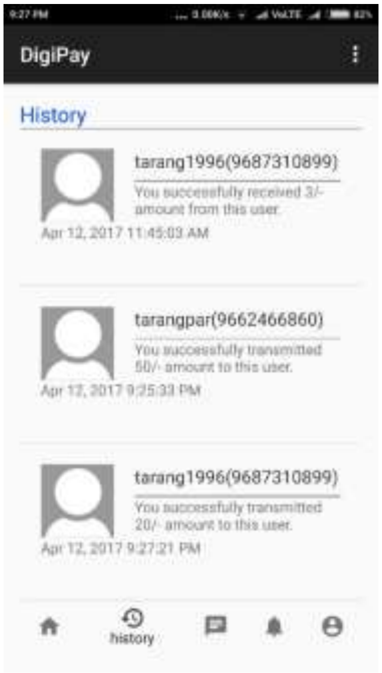


meet

Current Balance : 15.0

Transfer Done Successfully

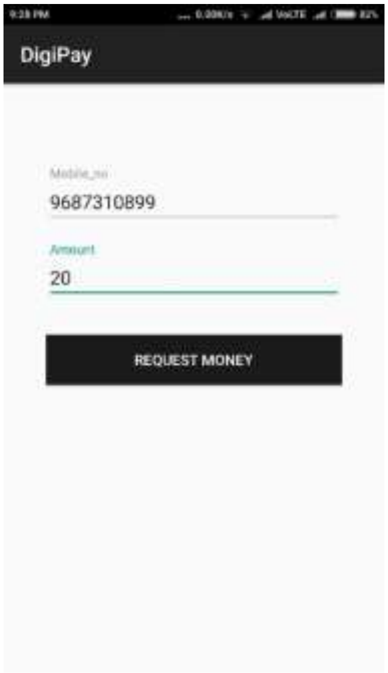
5. History Screen	**6. Request Screen**



tarang1996(9687310899)
You successfully received 3/- amount from this user.
Apr 12, 2017 11:45:03 AM

tarangpar(9662466860)
You successfully transmitted 50/- amount to this user.
Apr 12, 2017 9:25:33 PM

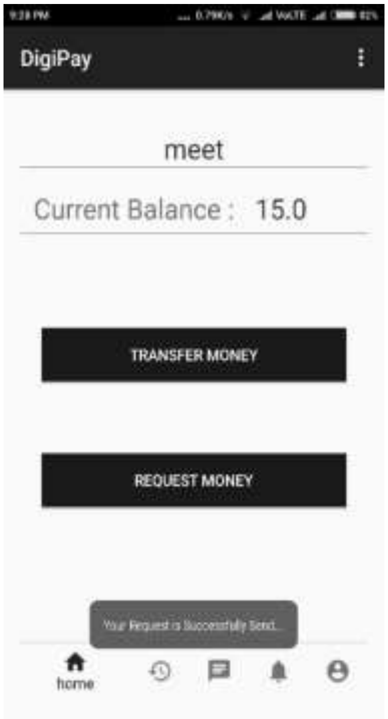
tarang1996(9687310899)
You successfully transmitted 20/- amount to this user.
Apr 12, 2017 9:27:21 PM



Mobile_no
9687310899

Amount
20

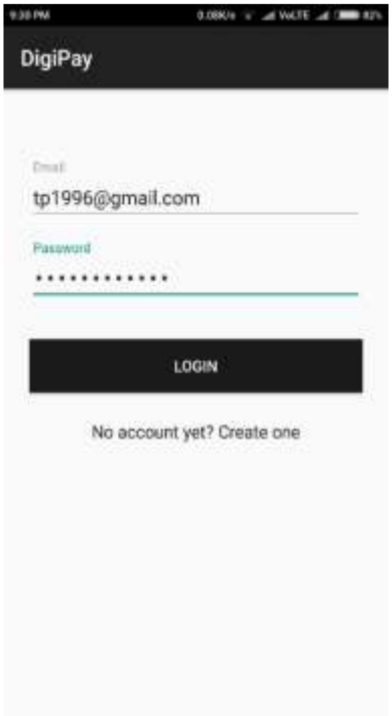
7.	**8. Login from Receiver Account**



meet

Current Balance : 15.0

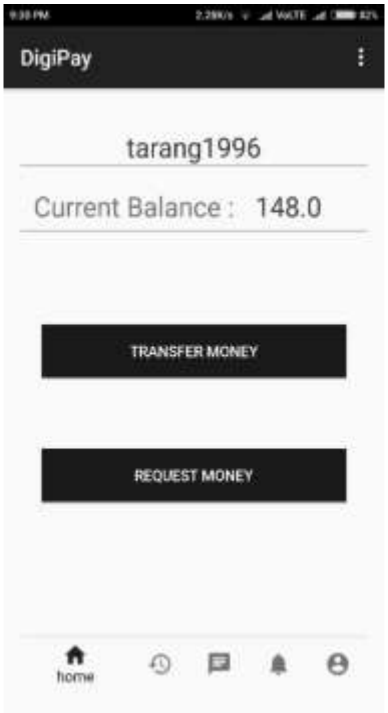
Your Request is Successfully Send...



Email
tp1996@gmail.com

Password
•••••••••••••

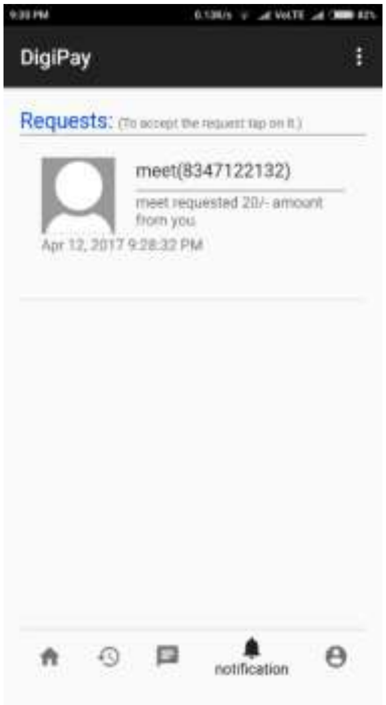
9. Welcome Screen



tarang1996

Current Balance : 148.0

10. Notification Screen



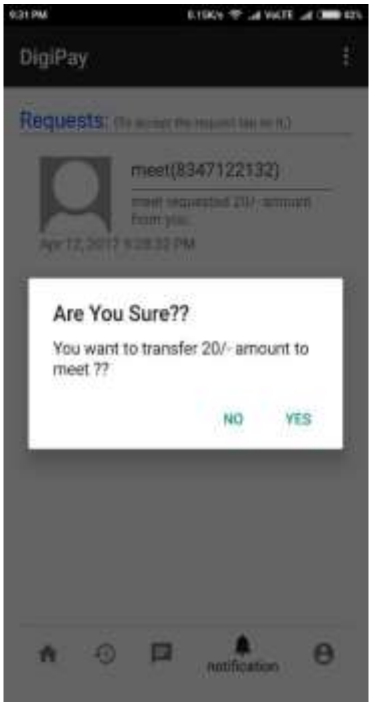
Requests: (To accept the request tap on it.)

meet(8347122132)

meet requested 20/- amount from you

Apr 12, 2017 9:28:32 PM

11.



Requests: (To accept the request tap on it.)

meet(8347122132)

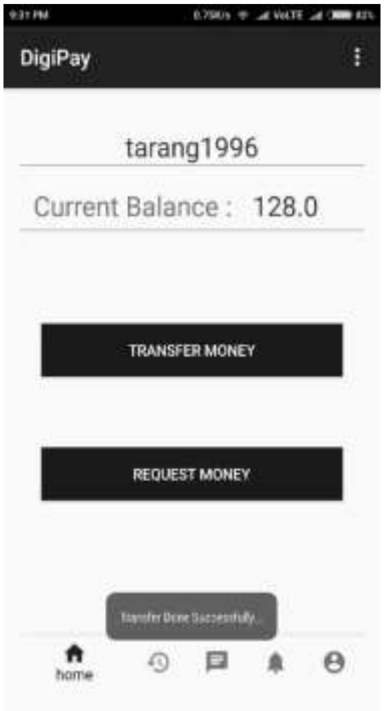
meet requested 20/- amount from you

Apr 12, 2017 9:28:32 PM

Are You Sure??

You want to transfer 20/- amount to meet ??

12. Request Complete

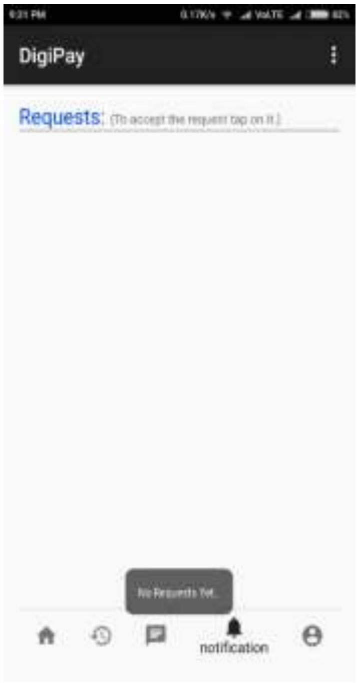


tarang1996

Current Balance : 128.0

Transfer Done Successfully..

13. Notification Screen

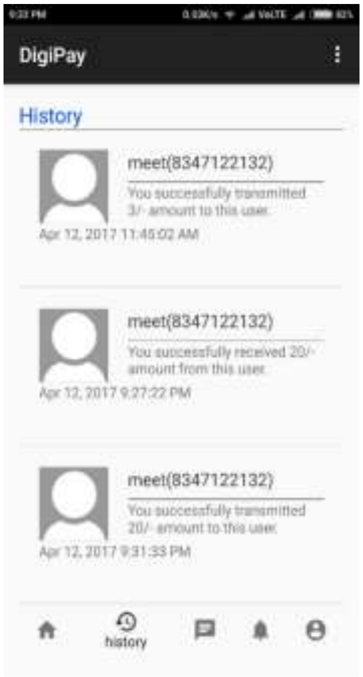


DigiPay

Requests: (To accept the request tap on it.)

No Requests Yet..

14. History Screen



DigiPay

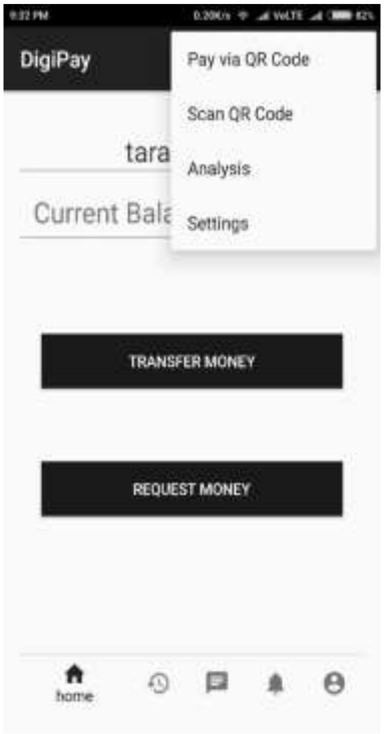
History

meet(8347122132)
You successfully transmitted 3/- amount to this user.
Apr 12, 2017 11:45:02 AM

meet(8347122132)
You successfully received 20/- amount from this user.
Apr 12, 2017 9:27:22 PM

meet(8347122132)
You successfully transmitted 20/- amount to this user.
Apr 12, 2017 9:31:33 PM

15. Other Option

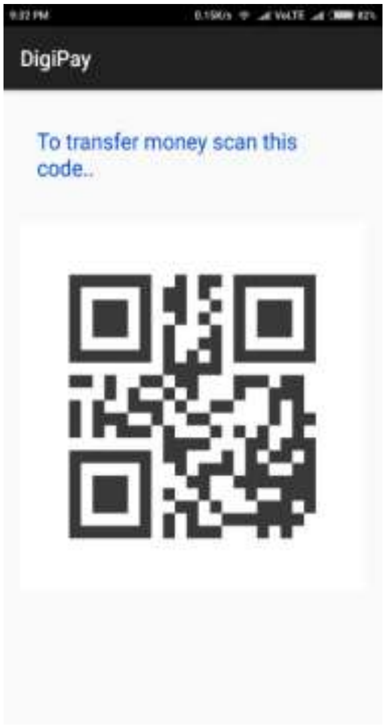


DigiPay

tara

Current Bala

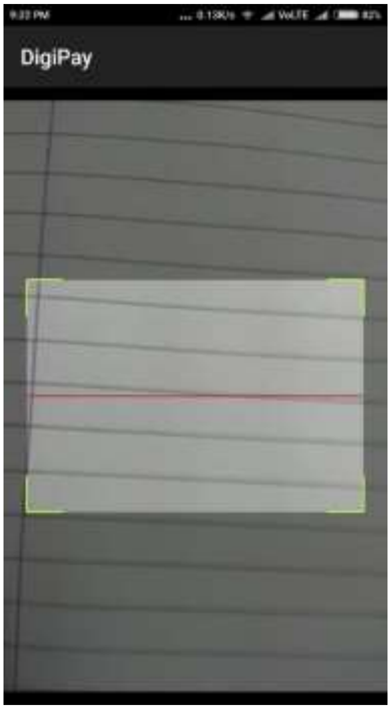
16. Generate QR Code

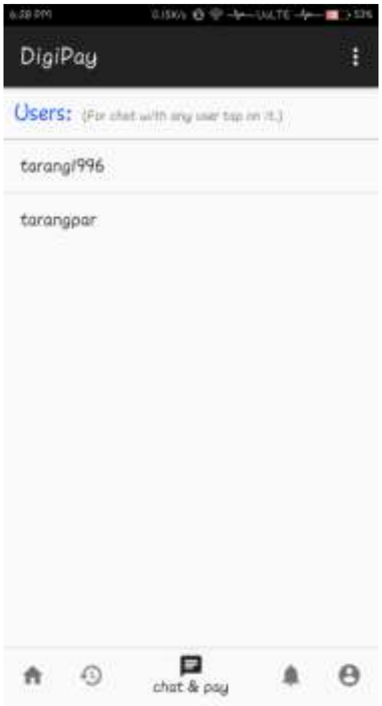


DigiPay

To transfer money scan this code..

17. Scan QR code and Transfer Particular Amount	18. Chat And Pay Screen



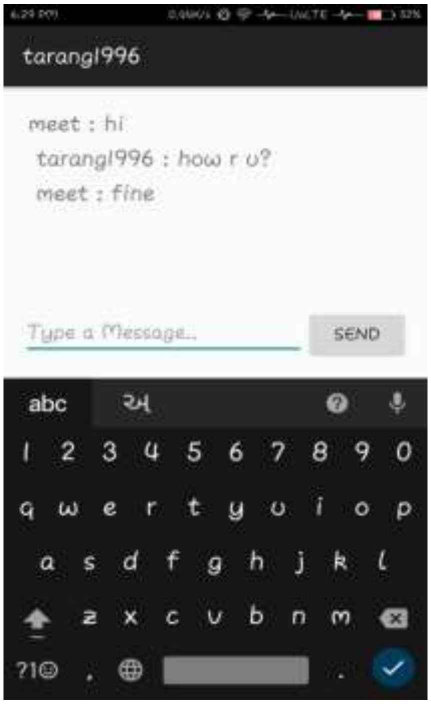


Users: (For chat with any user tap on it.)

tarangl996

tarangpar

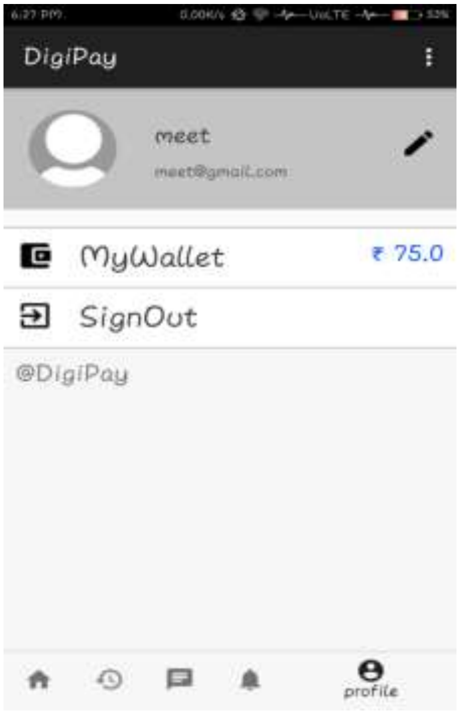
19. Send Message Screen	20. Profile Screen



meet : hi

tarangl996 : how r u?

meet : fine

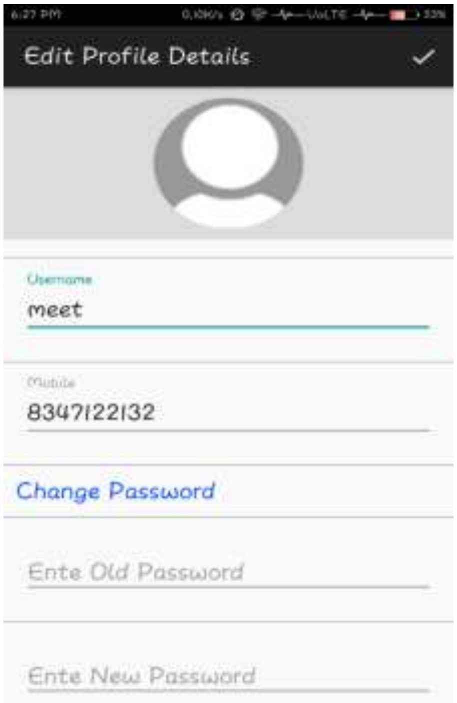


meet

meet@gmail.com

MyWallet ₹ 75.0

21.Edit Profile Details



Username

meet

Mobile

8347122132

Ente Old Password

Ente New Password

7. Limitations and Future enhancements

Limitations:

- By this application, we can’t use actual transaction

Future Enhancements:

- Various security features added for transaction
- More facility like video chatting and location of the people For that you transfer money

8. Conclusion

Hereby we declare that we had performed a project by understanding all module of this project. We checked the feasibility and requirement for this system. Then we defined overall look and flow of control among modules in the paper. After this, we started the actual design of our modules of the system in Java. All modules of the system were developed separately. Then we integrated all modules by means of control flow among all modules.

After Coding and integrating of all modules done, then we tested all modules separately this is basically Unit Testing of an all modules.by completion of Unit Testing, the whole system is then tested once again this called Integration Testing. Test cases were designed by performing black box testing.