A Project Report
On

BILLSPLIT

B. Tech (CE) Sem- VI

In fulfilment of all requirements for the subject of
SYSTEM DESIGN PRACTICE

Bachelor of Technology
In
Computer Engineering

Submitted by

Nikunj Dhola (CE-021)
(15CEUOS001)
Jaydeep Monapara (CE-064)
(15CEUOG098)

Under the Guidance of
Prof. Sidhdharth P. Shah

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

DHARAMSINH DESAI UNIVERSITY

NADIAD-387001, GUJARAT

CERTIFICATE

This is to certify that the project carried out in the subject of System Design Practice titled "BILLSPLIT" and recorded in this report is the bona fide work of

Nikunj S. Dhola	**CE-021**	**15CEUOS001**
Jaydeep V. Monapara	**CE-064**	**15CEUOG098**

of B. Tech, semester VI in the branch of Computer Engineering during the academic Session of December 2017 – April 2018.

Prof. Sidhdharth 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

Contents

ACKNOWLEDGEMENT

We have worked hard to bring this project to completion, however it would not have been possible without the kind support and help of many individuals. We would like to extend sincere thanks to all of them.

We are highly indebted to **Prof. Sidhdharth P. Shah** for his guidance and constant supervision as well as for providing constant information regarding project and other helpful guidelines.

We would also like to express our special gratitude and thanks to Head of Department **Dr C. K. Bhensdadia** for his kind support.

ABSTRACT

Nowadays, The most of payment among the groups of roommates, friends, family members or anyone are done by only one person out of them. Our application **BillSplit** takes the trouble out of sharing these type of shared expenses. The payer of expenses does not need to save the receipt of payment as he/she can add expenses as soon as they happen, in BillSplit. The trouble of remembering who, why and how much everyone owe in each shared expense is reduced by BillSplit. The data is stored " in the cloud ", so that anyone can access it anywhere on their phone.

Share bills and Make sure everyone gets paid back

CHAPTER 1
INTRODUCTION

- **Project Details: Broad Specification**

This project is aimed at developing an android based mobile application named BillSplit to make it easy to split bills with friends and family. The application organizes all shared expenses at one place, so that everyone can see who they owe. Whether anyone sharing a vacation, splitting rent with roommates, or owe someone for lunch, This application makes life easier. All the data in project is stored in cloud, so user can get access to them anywhere and anytime on their android devices. This project eliminates the need to keep the receipts of every shared expenses.

- **Technology Used:**

  - Front End: XML
    As the developing platform is Android, the technology used for the designing of front view is XML.

  - Back End: Java, JSON, Google Firebase
    The project is built using Java language and Google Firebase is used for database which uses JSON data format for manipulating data.

  - Diagram Tool: UMLet
    All the UML diagrams and various other diagrams corresponding to this project are made in UMLet.

CHAPTER 2

SOFTWARE REQUIREMENT SPECIFICATION

2.1 Purpose

The purpose of this document is to present a detailed description of the Billsplit. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed to the University for its Approval.

2.2 Document Conventions

The document is preparing using IEEE template of SRS (Ref: Copyright © 1999 by Karl E Wiegers). We describe requirements in natural language sentences. Functional requirements are described with input, output and processing.

2.3 Intended Audience and Reading Suggestions

This document is solely intended for study purpose and refers to subject Software Design Practice. It must be referred by the managing staff, developers, testers and documentation writers.

2.4 Product Scope

This system is designed for them who want to store the details of shared expenses among their groups and reduce the stress of collecting receipts of payments or registering individual's share in financial books.

- System provides interface to add and delete shared expenses with virtual money only.
- System does not have any value for the virtual money. This money does not have any monetary value.
- System allows to create group of friends, roommates, family members or anyone.
- System includes facility to add new friend.
- System provides interface to see all transactions which includes shares of that person.
- System does not include payment facility.

2.5 Product Perspective

This project is developed as part of course work in the subject of "Software Design Practice". The motive of this software is to provide an interface to every person who want to save shared expenses at one place eliminating need of any conventional method to store the details of these type of expenses.

2.6 Product Functions

- System provides interface to add new expense within groups or individual person with virtual money.
- System provides facility to track expenses to each user.
- System displays all past transactions within groups or individual person.
- System provides facility to pay all transactions' amount in one big payment(Settle Up), instead of bunch of small ones.
- System allows user to add new friend and create groups with their friends.
- System allows user to add new expense with various type of splitting options like Split Equally, Split by Exact Amount, Split by Percentage, Split by Share, Split by Adjustment.

2.7 User Classes and Characteristics

There is one type of user for this system:

(1) User :-

These users can add, delete and settle up shared expenses within their groups and with each other from this interface.

2.8 Operating Environment

<u>Sender and Receiver client:</u>

This application will only be available with Android Operating System. The application will only be used with compatible android devices. The user shall use this application on Android OS Jelly Bean 4.1(API 16) .

.

2.9 Design and Implementation Constraints

- The system allows all transaction based on only virtual money.
- The system does not allow users to do any type of real payments.

2.10 Assumptions and Dependencies

- System may not work if appropriate network or hardware is not supplied.

- Mainline network of system highly needs electricity to work on.

2.11 User Interfaces

Application can be accessed through only specific mobile interface. The software will be fairly compatible with android devices.

2.12 Hardware Interfaces

The application is intended to be a stand-alone, single-user system. The application will run on Android Devices or an Android Emulator. Any further hardware devices or interfaces will not required.

2.13 Software Interfaces

The application will run on the Android Operating System, specially version 4.1 (Jelly Bean) (API 16) and above.

2.14 Communications Interfaces

The P2P File Transfer System shall use the HTTP protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite.

2.15 System Features

2.15.1 System provides facility to the user to create his/her account

1.1 - System provides interface to the users to enter appropriate details.

Input: User Details

Output: User account

Description: The details include username, password, full name and mobile number. The system accepts email address as the username of the user.

2.15.2 System provides authentication facility to the user based on email ID and password.

2.1 - System authenticates user based on email ID and password.

2.1.1 - System provides interface through which user can enter email ID and password.

Input: Username and password.

Output: Authentication status.

Description: The system passes email ID and password to Firebase Authentication System to authenticate user. If the authentication is successful then successful login is provided to user otherwise If mismatch is found then error message will be displayed.

2.15.3 System provides facility to user to manage his/her profile.

3.1 - System displays the profile to the user.

3.1.1 - System shows the personal details of the user.

3.1.2 - System displays amount which is owed and amount which owes by him/her.

3.2 - System provides a facility to user to update his/her profile.

Input: Details which have to be modified.

Output: Updated profile

Description: The system allows users to modify the details entered at the time of creating their accounts and saves updated details in database.

3.3 - System provides a facility to user to update his/her profile picture.

3.3.1 - System opens the gallery of the user device to select image.

3.3.2 - System provides interface to crop the selected image as per the need of the user.

3.3.3 - System saves the updated profile picture in Firebase.

2.15.4 System provides facility to view relative amount with all friends and settle up with them.

4.1 - System displays relative amounts with all friends.

4.2 - System allows user to settle up with particular friend.

Description: The System show a dialogue box asking for the confirmation of settle up. If user allows then the relative amount will be settled up.

2.15.5 System provides facility to see the list of all transaction and details of particular transaction and to delete transaction to the user.

5.1 – System displays the list of all past transactions made with his/her friends and groups.

5.2 - System displays details of the transaction from the past transaction list.

Description: The detail of the transaction includes description, each user's share, payer's information, date and total amount of transaction.

5.3 - System provides the facility to delete the transaction from the list of past transactions.

Input: Transaction have to be deleted.

Output: Updated Transaction List

2.15.6 System provides facility to add new expense.

6.1 - System allows user to add new expense splitting equal share of total amount with individual friend.

Input: Id of the user along with the details of transaction

Output: Relative amount is added into the user and friend's balance

Description: The user select friend from its list then system provides interface to enter the description, total amount and date of the transaction. The equal share of total amount is added to both user and friend's balance.

6.2 - System allows user to add new expense in group.

6.2.1 - System provides interface to enter description, date, total amount and suggestion of group names to be selected.

6.2.2 - System provides options for splitting the expense by Exact Amount, Share, Percentage, Adjustment or Equally.

6.2.3 - As per selected splitting option, the system provides interface to collect the individual's share in the transaction.

6.2.4 - The transaction is saved in database and displayed in past transactions list.

2.15.7 System provides facility to add new friend and create new group.

7.1 - System allows user to add new friends.

Description: System displays all users' email addresses. System also provides facility to search user by email address. User selects one of them to add as a new friend. Then the system adds selected user into the friendlist.

7.2 - System allows user to create new group.

7.2.1 - System provides interface to enter the name of new group.

7.2.2 - System displays all users' email addresses and also provides facility to search user by email address.

7.2.3 - User selects all users who will be the members of new group.

7.2.4 - On the confirmation of user, the group is created with selected users.

2.16 Other Non-Functional Requirements

Safety Requirements

Information transmission should be securely transmitted to server without any changes in information.

Security Requirements

For functional requirements like user log in the system, there should be mechanism to prevent any type of interference with transmitted information.

2.16.1 Performance

The system must be interactive and the delays involved must be less. So in every action-response of the system, there are no immediate delays. In case of opening any activities, of popping error messages and saving the settings there is delay much below 2 seconds, in case of opening databases, sorting questions and evaluation there are no delays and the operation is performed in less than 2 seconds for opening, sorting, computing, posting > 95% of the files. Also, when connecting to the server the delay is based editing on the distance of the 2 systems and the configuration between them so there is high probability that there will be or not a successful connection in less than 20 seconds for sake of good communication.

2.16.2 Safety

Information transmission should be securely transmitted to server without any changes in information

2.16.3 Reliability

As the system provide the right tools for discussion, problem solving it must be made sure that the system is reliable in its operations and for securing the sensitive details.

2.16.4 Availability

If the internet service gets disrupted while sending information to the server, the information can be send again for verification.

2.16.5 Security

The main security concern is for users account hence proper login mechanism should be used to avoid hacking. The tablet id registration is way to spam check for increasing the security. Hence, security is provided from unwanted use of recognition software.

2.16.6 Usability

As the system is easy to handle and navigates in the most expected way with no delays. In that case the system program reacts accordingly and transverses quickly between its states.

CHAPTER 3
DESIGN

3.1 Use Case Diagram

- Use-case diagram for User

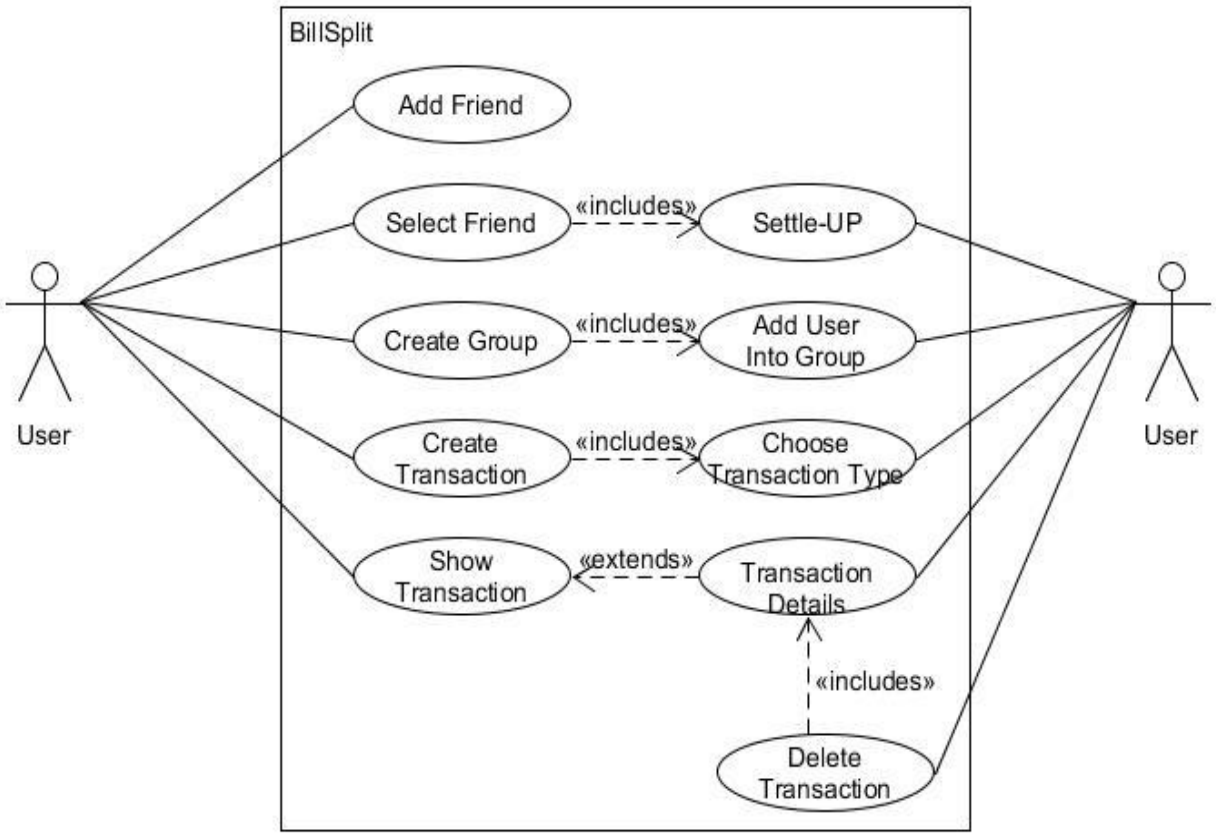


Fig 3.1 Use Case Diagram for USER of BillSplit

3.2 Class Diagram

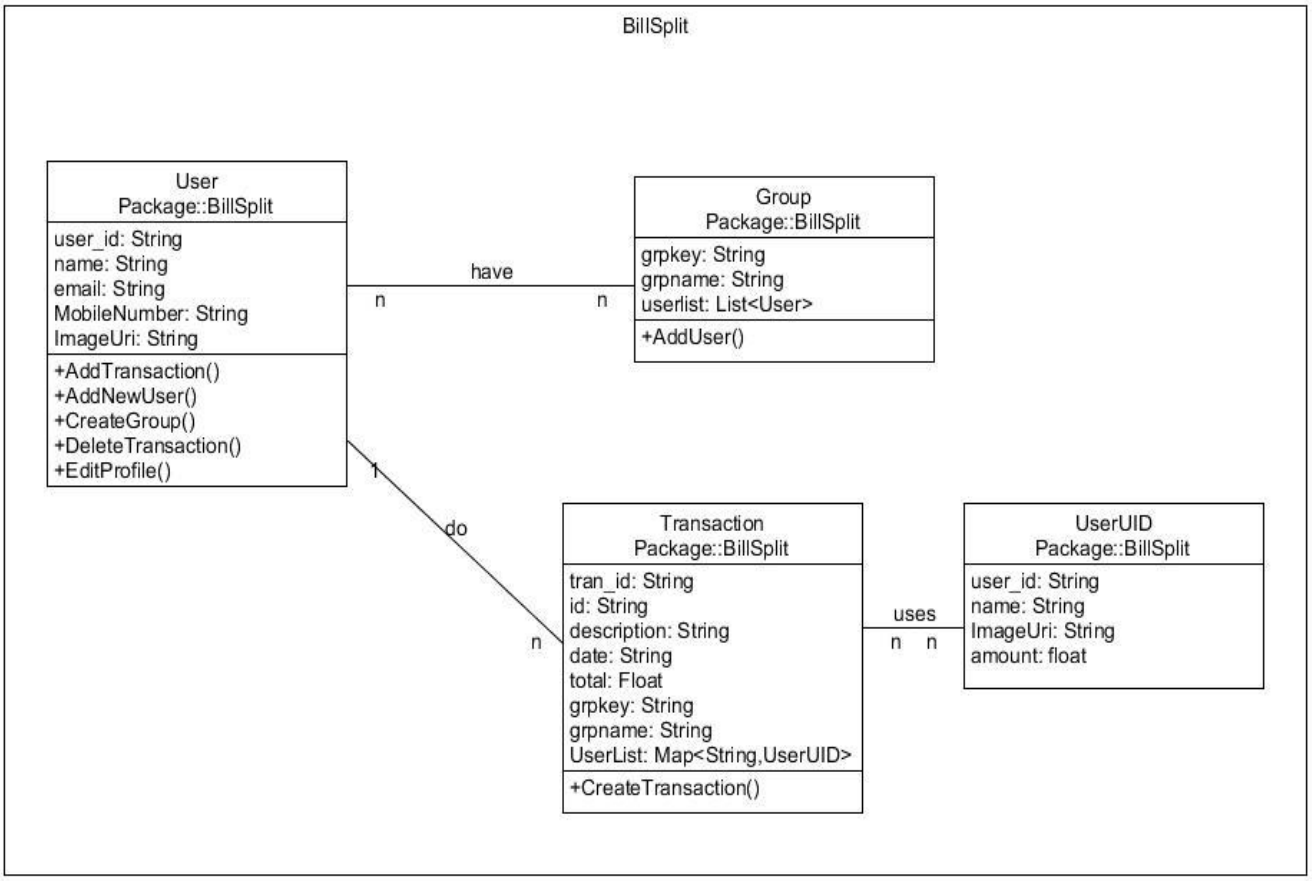


Fig 3.2 Class Diagram for BillSplit

3.3 Sequence Diagram

- Sequence Diagram for buying stocks

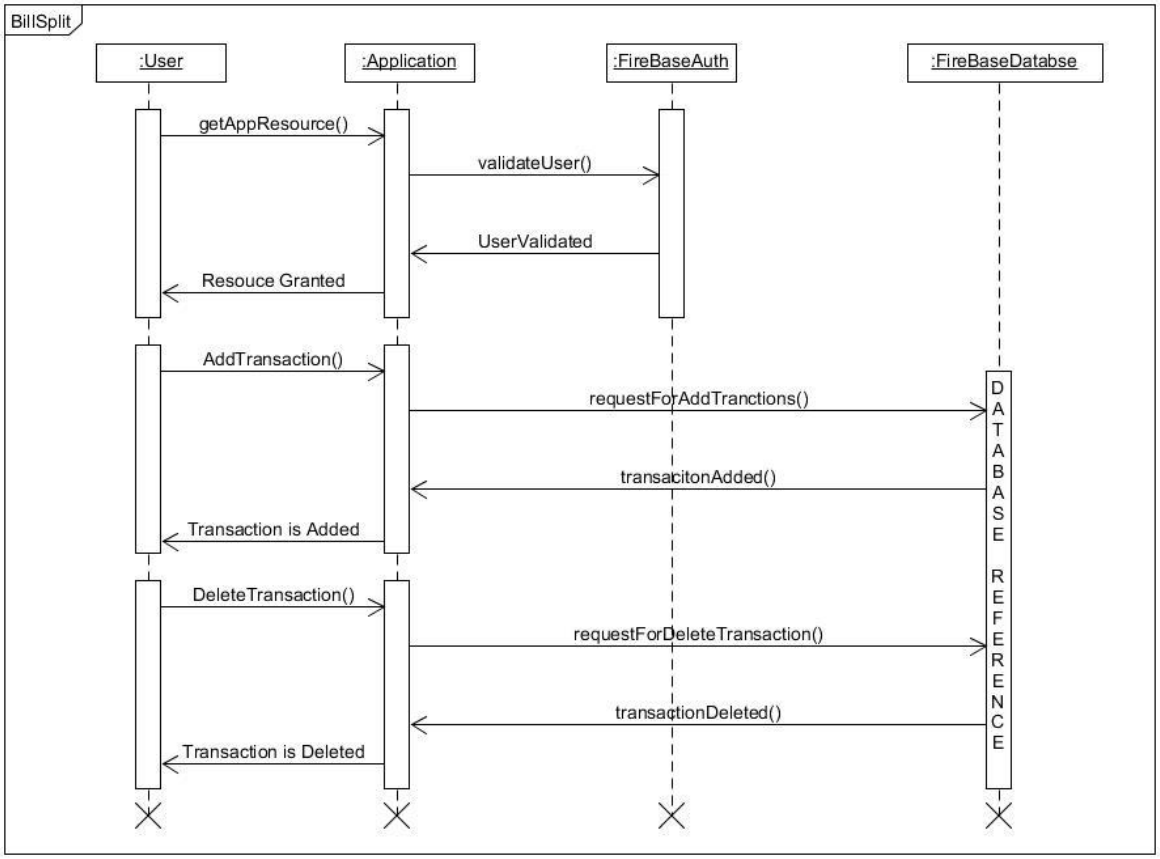


Fig 3.3 Sequence Diagram to ADD & DELETE Transaction

3.4 Activity Diagram

- Activity diagram for Add Transaction

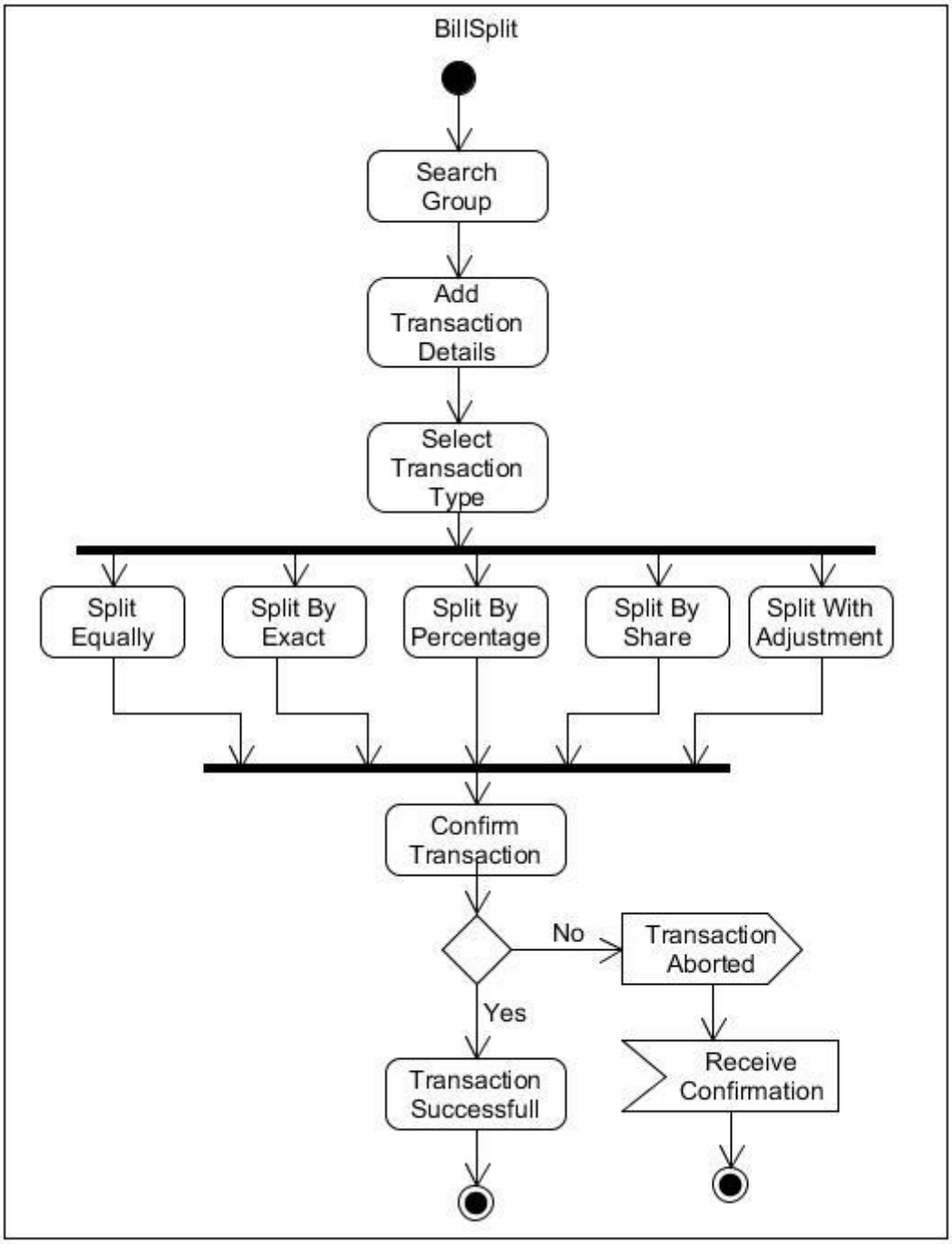


Fig 3.4 Activity Diagram for ADD New Transaction

- Activity diagram for Delete Transaction and Create New Group

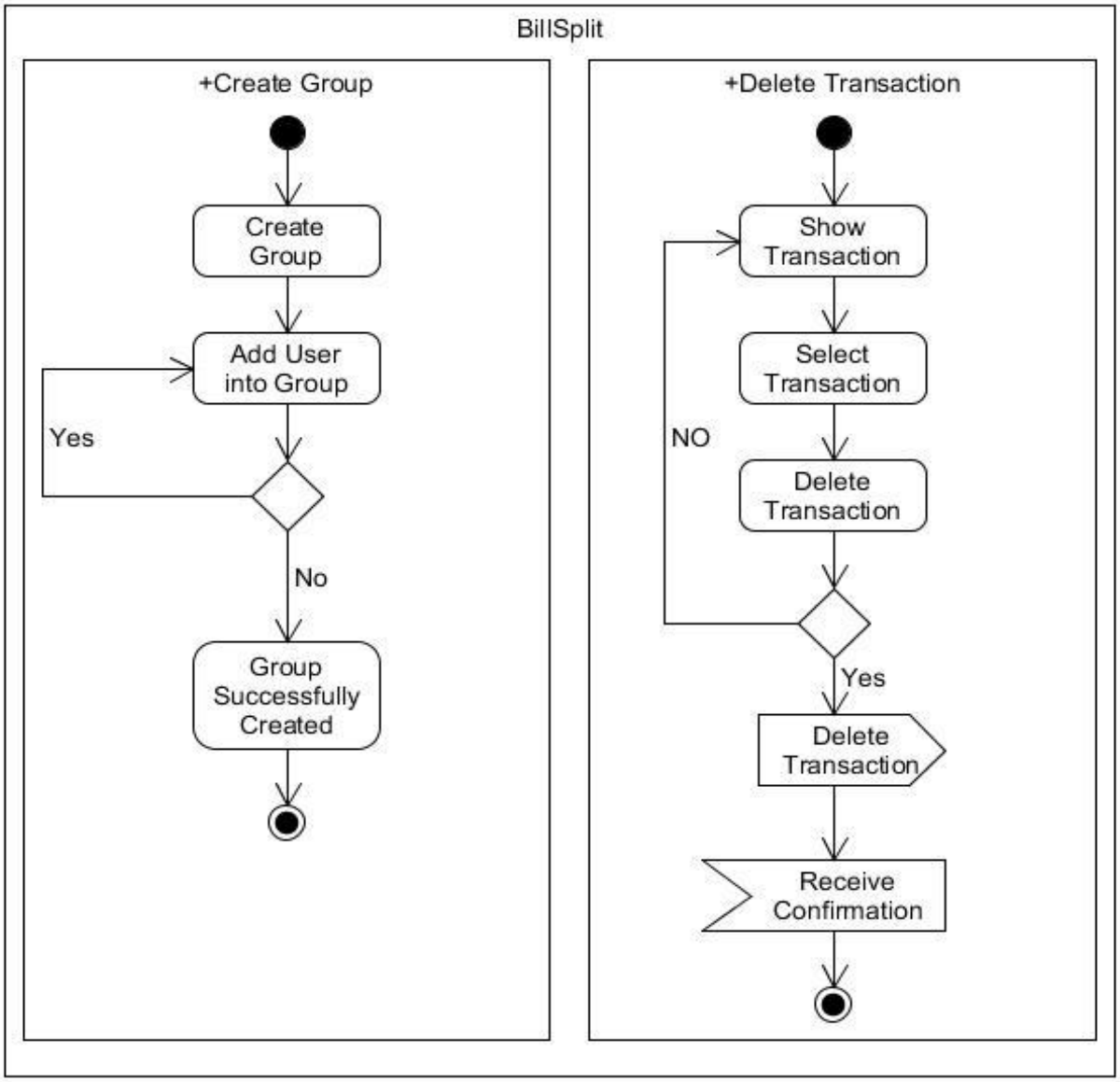


Fig 3.5 Activity Diagram for DELETE Transaction and CREATE New GROUP

CHAPTER 4
IMPLEMENTATION DETAILS

4.1 Implementation Environment

- Android Studio 3.0.1
- Google Firebase 11.8.0

4.2 Module Description

- ### Login Module :

User provides his/her login details like Email Id and Password. The login module passes credentials to Google Firebase Authentication System which checks for user validation. If the user credentials match then user will be provided successfully login otherwise the error message will be displayed.

- ### Signup Module :

User provides registration details including Full Name, Mobile Number, Email Id and Password. Then Sign Up module passes Email Id and Password to Firebase Signup Authentication System which registers user. After successful registration Login Module will be provided.

- ### Add New Bill Module :

User provides details of expense which includes Description, Total Amount, Date and Group Name or individual friend's details. User will choose the suitable Split Type from Split Equally, Split by Exact Amount, Split by Percentage, Split by Shares and Split by Adjustment. After that the transaction will be performed by module.

- Delete Transaction Module :

User is provided the list of tansactions which are done within its group or individually with friends. Then user chooses the transaction to be deleted from the list. Transaction details is provided to user. After reviewing details user can delete transaction.

- Create New Group Module :

User enters the group name. Then the list of all users is provided by module. The user searching facility is also provided. User selects group members by email id from the list. Then the group is created.

CHAPTER 5

TESTING

5.1 Black Box Testing

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.

Test Case ID	Test Scenario	Test Data	Expected Result	Actual Result
T01	User Login	Correct Username and Password	Login successful and redirect to HomeActivity	As expected
T02	User Login	Correct Username and incorrect Password	Login Fails and Alert Message shown	As expected
T03	User Login	Incorrect Username	Login fails and Alert Message shown	As expected
T04	User Signup	Valid Registration Details	Signup successful	As expected
T05	User Signup	Invalid Registration Details	Signup failed	As expected
T06	Add Bill (Split Equally)	Details of Expense (At least one user is involved)	Bill Added Successfully	As expected
T07	Add Bill (Split Equally)	Details of Expense (No participant)	Error Message displayed	As expected
T08	Add Bill (Split By Exact Amount)	Details of Expense (Sum of individual's amount Equal to total amount)	Bill Added Successfully	As expected

T09	Add Bill (Split By Exact Amount)	Details of Expense (Sum of individual's amount Not Equal to total amount)	Error Message displayed	As expected
T10	Add Bill (Split By Percentage)	Details of Expense (Sum of individual's Percentage Equal to 100)	Bill Added Successfully	As expected
T11	Add Bill (Split By Percentage)	Details of Expense (Sum of individual's Percentage Not Equal to 100)	Error Message displayed	As expected
T12	Add Bill (Split By Share)	Details of Expense (Total share more than zero)	Bill Added Successfully	As expected
T13	Add Bill (Split By Share)	Details of Expense (Total share Equal to zero)	Error Message displayed	As expected
T14	Add Bill (Split By Adjustment)	Details of Expense(Total Adjustment less or equal to Total Amount)	Bill Added Successfully	As expected
T15	Add Bill (Split By Adjustment)	Details of Expense(Total Adjustment more than Total Amount)	Error Message displayed	As expected
T16	Add Bill	Details of Expense (Description or Amount is null)	Error Message displayed	As expected
T17	Delete Expense	Key of Expense to be deleted	Expense Deleted Successfully	As expected

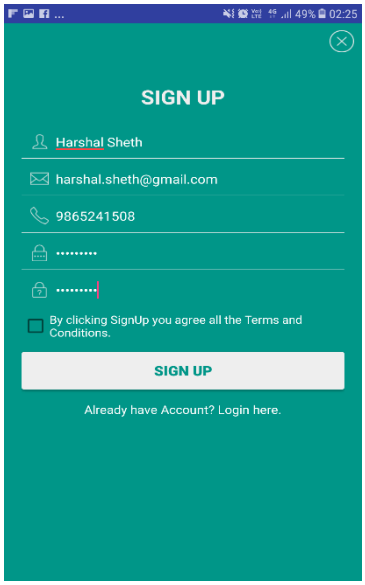
T18	Settle Up	Key of friend	Settled up Successfully	As expected
T19	Add New Friend	Key of User	User is Successfully Added as friend	As expected
T20	Add New Friend	Key of User (Already in Friendlist)	Error Message displayed	As expected
T21	Create New Group	Name of Group and Key of Members	Group is created Successfully	As expected

CHAPTER 6

SCREENSHOTS

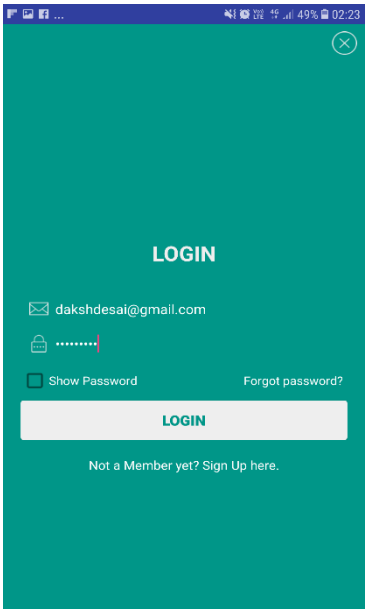
- ## Register User

  User has to register first into the system with his Username, Password, Full Name and Mobile Number.



- ## Log in

  User has to login into the system with his Username and Password.

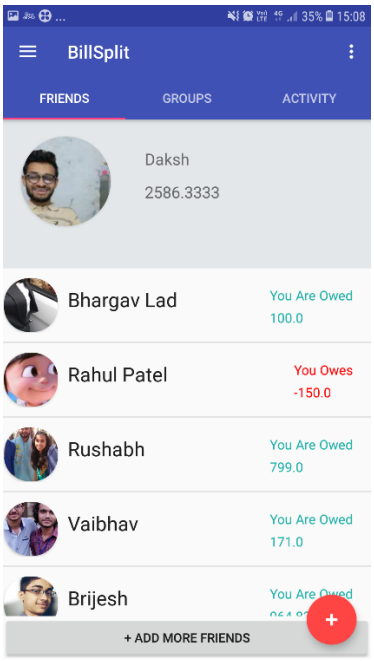


- ## User Home Activity

  The Home Activity shows three tabs and one floating button to add new group expense.

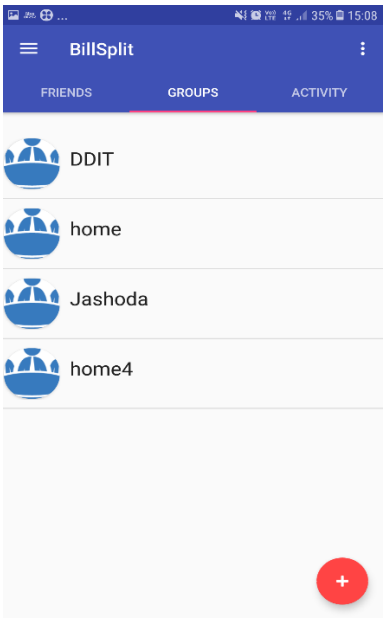
  ### 1. Friends Tab

  It shows the list of friends of user and their relative amounts. It also shows User's Total Balance.



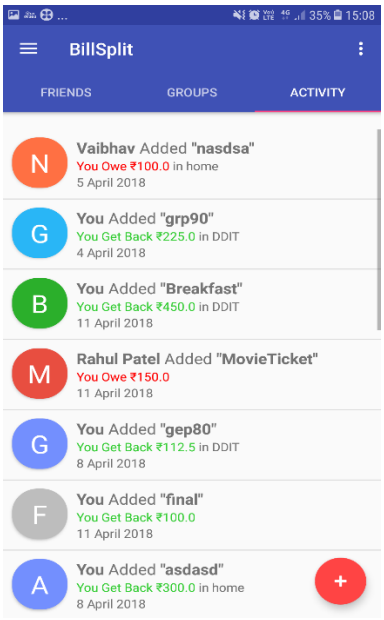
  ### 2. Friends Tab

  It shows the list of groups of user.



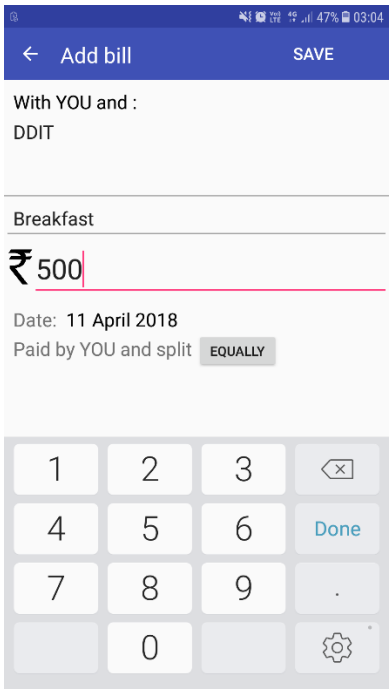
3. Activity Tab

It shows the list of all past expenses of user within groups and with individual friend.



- **Add new expense**

To add new expense, user has to enter description, total amount of expense and select date and suitable splitting option by clicking button with text Equally.

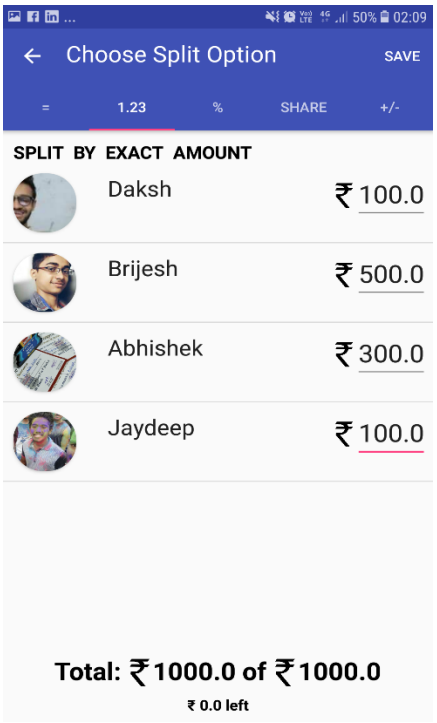


- ### Selecting Suitable Splitting Option

  Five tabs are provided.

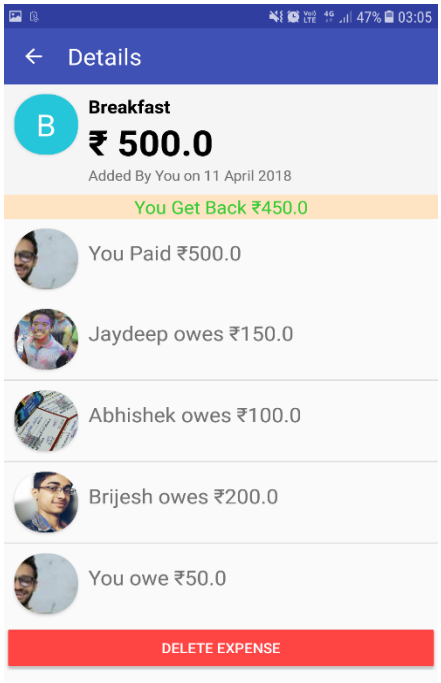
  1. Split Equally
  2. Split By Exact Amount
  3. Split By Percentage
  4. Split By Share
  5. Split By Adjustment

  The following figure shows Split By Exact Amount tab which provide interface to enter the exact amount of the participant of expense.



- ## View Details of Expense

  The details of particular expense is shown with Description of Expense, Date, Payer Information and share of each participant and delete button to delete expense.



CHAPTER 7
CONCLUSION

We wanted to learn Android App Development and the best way to learn anything is to start a project in it. Meanwhile we also wanted an interface for Storing Shared Expenses so we choose this subject as our project definition and have learned and explored a lot about these topics.

Here by We declare that we developed a project by understanding all modules of it. We gathered requirement for this system. Then we defined overall control flow among modules. After this, We started coding our modules. All modules of system were developed separately. Then we integrated all modules.

After the development of all modules, We tested modules separately i.e. unit testing, after integrating modules, Integration Testing was performed. Test cases were designed using black box testing technique.

7.1 Limitations

- This System does not provide actual payment facility.
- This system does not provide detailed description of any expense.

7.2 Future Enhancements :

- To provide actual payment facilities through UPI, Paytm, etc.
- To provide facility of More Than One Payer in shared expenses.
- Editing of expenses.
- To provide facility to upload image of payment receipt.

CHAPTER 8

SOFTWARE VERSION DEPLOYMENT

8.1 Installation Steps

- Download Android Studio.
  Download the Linux SDK from the Android website.

- Run the following commands.

```
sudo apt-get install unzip
sudo tar xvzf android-studio-ide-135.1641136-
linux.zip
cd android-studio-ide-135.1641136-linux
./studio.sh
```

- Sync Project With Google Firebase.

8.2 Deployment Steps

- Connect to Firebase.
- Click **Tools > Firebase** to open the **Assistant** window.
- Click to expand one of the listed features (for example, Analytics), then click the provided tutorial link (for example, Log an Analytics event).
- Click the **Connect to Firebase** button to connect to Firebase and add the necessary code to your app.

CHAPTER 9
BIBLIOGRAPHY

Referred Websites:

- www.developer.android.com
- www.firebase.google.com
- www.stackoverflow.com
- www.android.com
- www.androidhive.info