Project Report on

GROUP EXPENSE TRACKER Application

ASISHKUMAR GOUDA

(2106827)

Project partner

YASH SAVLA

(2106871)

Guided By

Mrs. PRAJAKTA KAMBLE

Academic Year 2021-2022

K.J. Somaiya College of Science and Commerce

K.J. Somaiya College of Science and Commerce

Vidhyavihar, Mumbai - 400 077

Re-accredited with 'A' Grade by NAAC

Autonomous - Affiliated to University of Mumbai

Department of Information Technology

<u>CERTIFICATE</u>

Certified that the experimental work was entered in this journal as per syllabus in **B.Sc. Information Technology** for **Software Project Management** as prescribed by University of Mumbai and was done in K.J. Somaiya College of Science and Commerce by the student Shri <u>**ASISHKUMAR SARATKUMAR GOUDA**</u> having Seat No. <u>**2106827**</u> Of class **TY B.Sc. Information Technology** during the academic year 20**21** - 20**22**

Course Coordinator

Date

Sign of In-charge   College Seal   Sign of Examiner

Date: Date:

Acknowledgement

In the present world of competition there is a race of existence in which those are having will come forward succeed. Project is like a bridge between theoretical and practical working.

We express our deep gratitude and appreciation to those who agreed to participate in this project, for their time expended and courage in sharing their insights with the fledging students. It is to them that we are most indebted, and we can only hope that the product of our collaboration benefits each one as much as we are benefitted from process.

We have been immeasurably enriched by working under the supervision of Mrs. Prajakta Kamble, the subject teacher, who has a great knowledge and who has an art of encouraging, correcting and directing us in every situation possible, which has enabled us to complete our project.

At times, our studies carried out at great cost to those closest to us. We thank our family and our fellow students for their best understanding and support. We acknowledge to all the people who have involved and supported us in making this project.

TABLE OF CONTENTS

Abstract

This Expense Tracker application falls in the Finance Category and serves the important purpose of managing finances which is a very important part of one's life. This application allows multiple or single users to maintain a digital automated diary. User need to register first to access the application for which the user will be provided an id, which will be used to maintain the record of each unique user.

Application allows user to add trip to calculate all the expenses done by each member in the group. Expense Tracker application which will keep a track of Income-Expense of a user on a day-to-day basis. This application takes all the expenses from multiple users and divides the amount round sum amount between each.

Expense tracking application will generate report as and when required to show the expense via multiple graphs. The application also specifies the name of group member to whom the amount is to be given equally.

Chapter 1

- # Introduction

The purpose of this document is to collect, analyse and high-level requirements and features of this mobile application. It focuses on the capabilities needed by the target users. The document contains a mention of the purpose of the system, its overall description, specific requirements pertaining to the same and other supporting information.

- # Background

The group expense tracker manages all your financial accounts in one convenient place and makes your money management much easier and handy. Our expense tracking software lets you compare your spending with other friends who share similar group circles so you can see how much they typically spend and identify the best ways to save on your money. With expense tracker you can not only view your investments at each stage but also can send notifications to Facebook friends at the time of group creation and logging expenses. Our application is new concept very easy and simple to handle, not very complicated as other mobile applications. The main benefit of using mobile phone is that it's simple and innovative.

Chapter 2

SURVEY OF TECHNOLOGIES

Survey of technology

XML: Extensible Markup Language (XML) is a markup that defines a set of rules for encoding documents in a format that is both human readable and machine readable The World wide web consortium's XML 1.0 specification of 1998 and several other related specifications all of them free Open Standards - define XML.

JAVA: Java is a high-level programming language originally developed by sun Microsystems and released in 1995. Java is a portable Language which runs on variety of platforms Such as windows, Linux, Mac Os, etc. It provides multiple features like much more compatible, stable, secure, etc. It also supports multi- Tasking.

Android: Android Language was developed in 2008 which uses Java as Programming Language and also uses Kotlin. Google states that "Android apps can be created by using Kotlin, Java and C++"but most of the android apps can be created using Java and it is written using Android (SDK)

SQLite Database: SQLite is an open-source Relational Database i.e., used to perform database operations on the android devices such as storing manipulating or retrieving Persistent data from the database. It is embedded in Android by default So there is no need to perform any database setup or administration tasks.

Chapter 3

REQUIEMENTS AND
ANALYSIS

Requirements and Analysis

3.1 Problem Definition

This Group Expense Tracker application falls in the Finance Category and serves the important purpose of managing finances in a group which is a very important during a trip. This application allows multiple or single users to maintain a digital automated diary. User need to register first to access the application for which the user will be provided an id, which will be used to maintain the record of each unique user. Application allows user to add trip to calculate all the expenses done by each member in the group. Group Expense Tracker application which will keep a track of Income-Expense during a trip. This application takes all the expenses from multiple users and divides the amount, round sum amount between each person in a group. Group Expense tracking application will generate report as and when required to show the expense via multiple graphs. The application also specifies the name of group member to whom the amount is to be given equally.

3.2 Requirement specification

- ## Functional Requirements:
- **User can create account**
- **Register/Login**
- **Log expenses**
- **Notification**
- **Report**

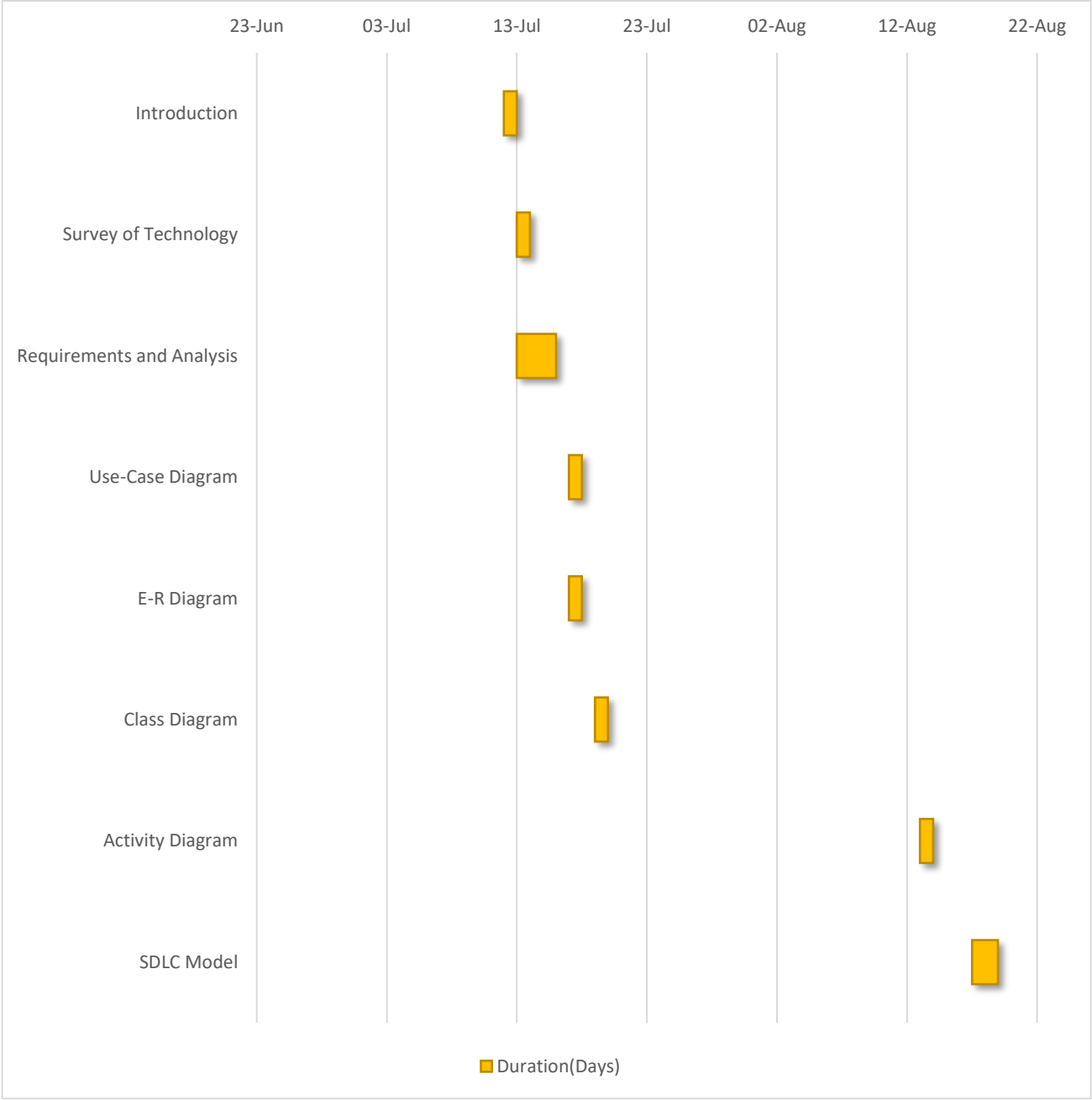## ➢ Non-Functional Requirements:

- **Availability**:
  The app service must be made available 24/7.
- **Security**:
  The app must encrypt user passwords and must assures all the security aspects of user personal data.
- **Maintainability**
  The app should be updated on the regular basis so that solution to any issues or new features will be incorporated.
- **Reliability**:
  The app must never crash more than 10 minutes per month and mut be trustworthy.
- **Design Constraints:**
  The app design must give an attractive look.

3.3 Planning and Scheduling

Task	Assigned to	Start date	End date	Status
Introduction	Asishkumar	12/07/2021	12/07/2021	Completed
Survey of Technology	Yash	12/07/2021	12/07/2021	Completed
Requirements and Analysis	Both	13/07/2021	15/07/2021	Completed
Use-Case Diagram	Asishkumar	17/07/2021	17/07/2021	Completed
E-R Diagram	Yash	17/07/2021	17/07/2021	Completed
Class Diagram	Yash	19/07/2021	19/07/2021	Completed
Activity Diagram	Asishkumar	13/08/2021	13/08/2021	Completed
SDLC Model	Yash	17/08/2021	20/08/2021	Completed

Gantt Chart

Chart axis dates: 23-Jun, 03-Jul, 13-Jul, 23-Jul, 02-Aug, 12-Aug, 22-Aug

Tasks:
- Introduction
- Survey of Technology
- Requirements and Analysis
- Use-Case Diagram
- E-R Diagram
- Class Diagram
- Activity Diagram
- SDLC Model

Legend: Duration(Days)

3.4 Hardware and Software requirements

<u>Hardware specification: (Server Side)</u>

For Development of this application, we have used laptop because it can handle multiple emulators for the better output.

Full Specifications:
- Processor: Intel(R) Core (TM) i5-8250U CPU @ 1.60GHz
- RAM: 8.00 GB
- Graphics: GPU Radeon (TM) 520 Discrete/Hybrid
- HDD Drive: 1TB
- SSD drive: 128 GB

For the testing of the application on a real device we have used an Android Phone

<u>Hardware specification: (Client Side)</u>

To run the Application, the user must have these software specifications.

Minimum Requirements:

- Processor: 1.25 GHz
- RAM: 2GB
- Storage: 500MB

Recommended Requirements:

- Processor: 2.00 GHz
- RAM: 4GB
- Storage: 1GB

Software specification: (Server Side)

For Development of this application, we have used laptop because it can handle multiple emulators for the better output.

- Android Studio (latest version above 3.5)
- Front End: XML
- Backend: Java, SQLite
- Programming: Android

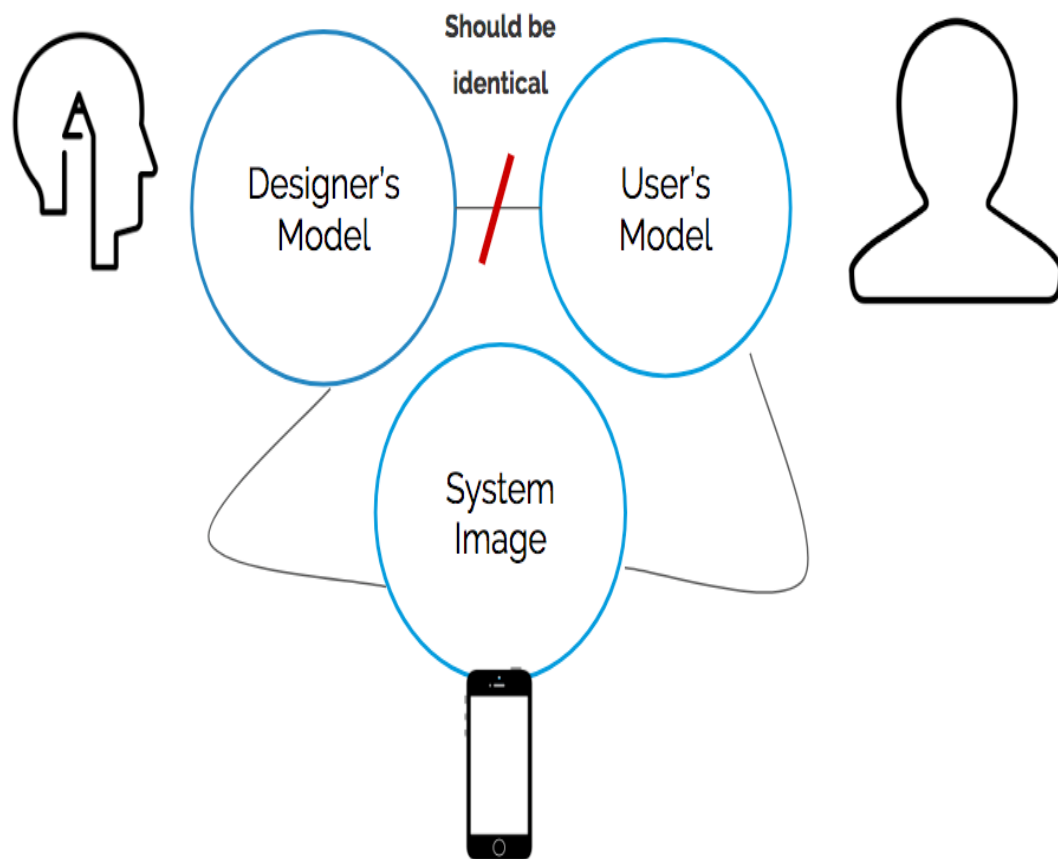## **Software Specifications: (Client Side)**

Minimum Requirements:

- Operating System: Android
- Android Version: Ice-Cream Sandwich v4.0

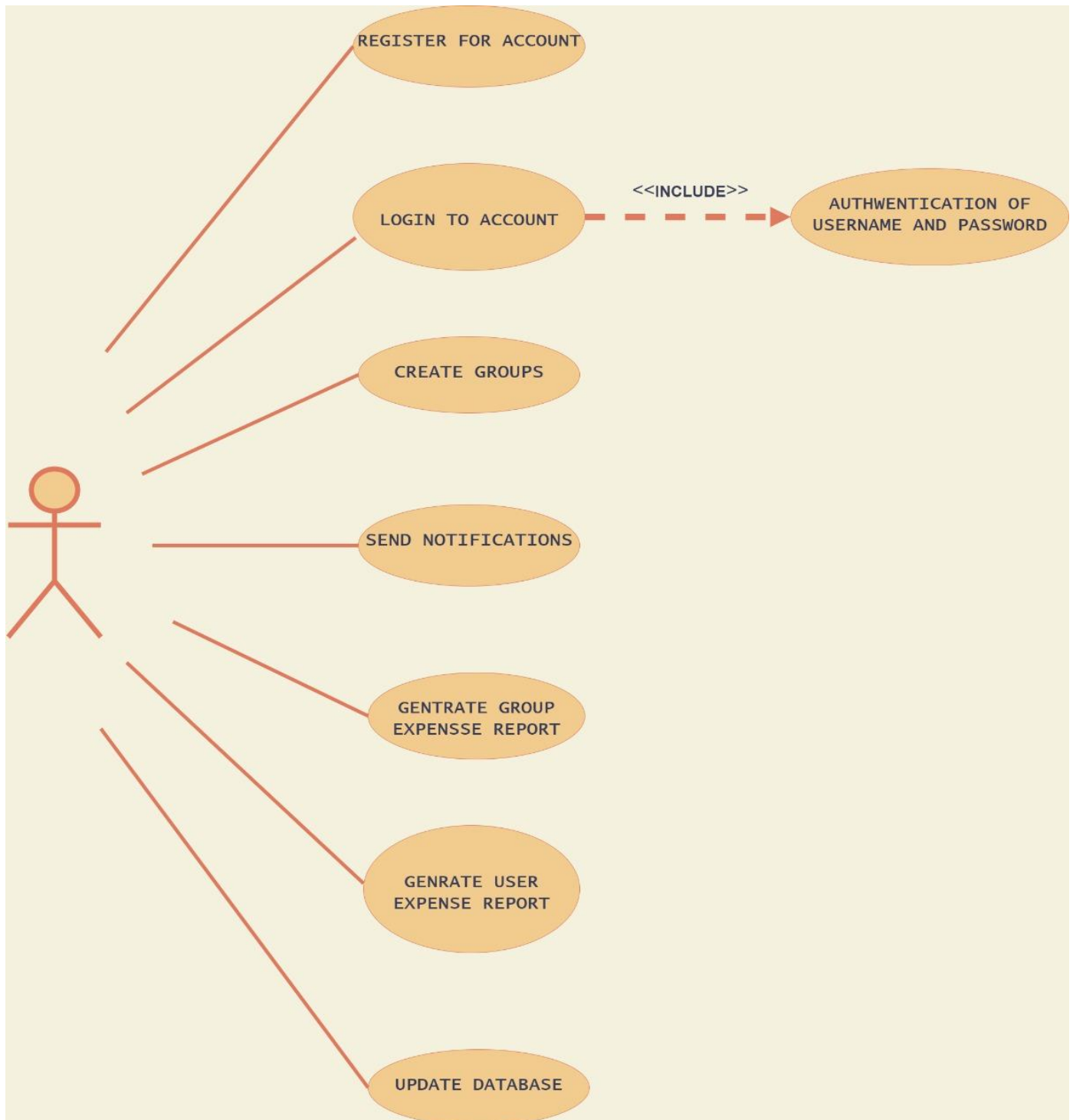## **Recommended Requirements:**

- Operating System: Android
- Android Version: Android Q v10.0
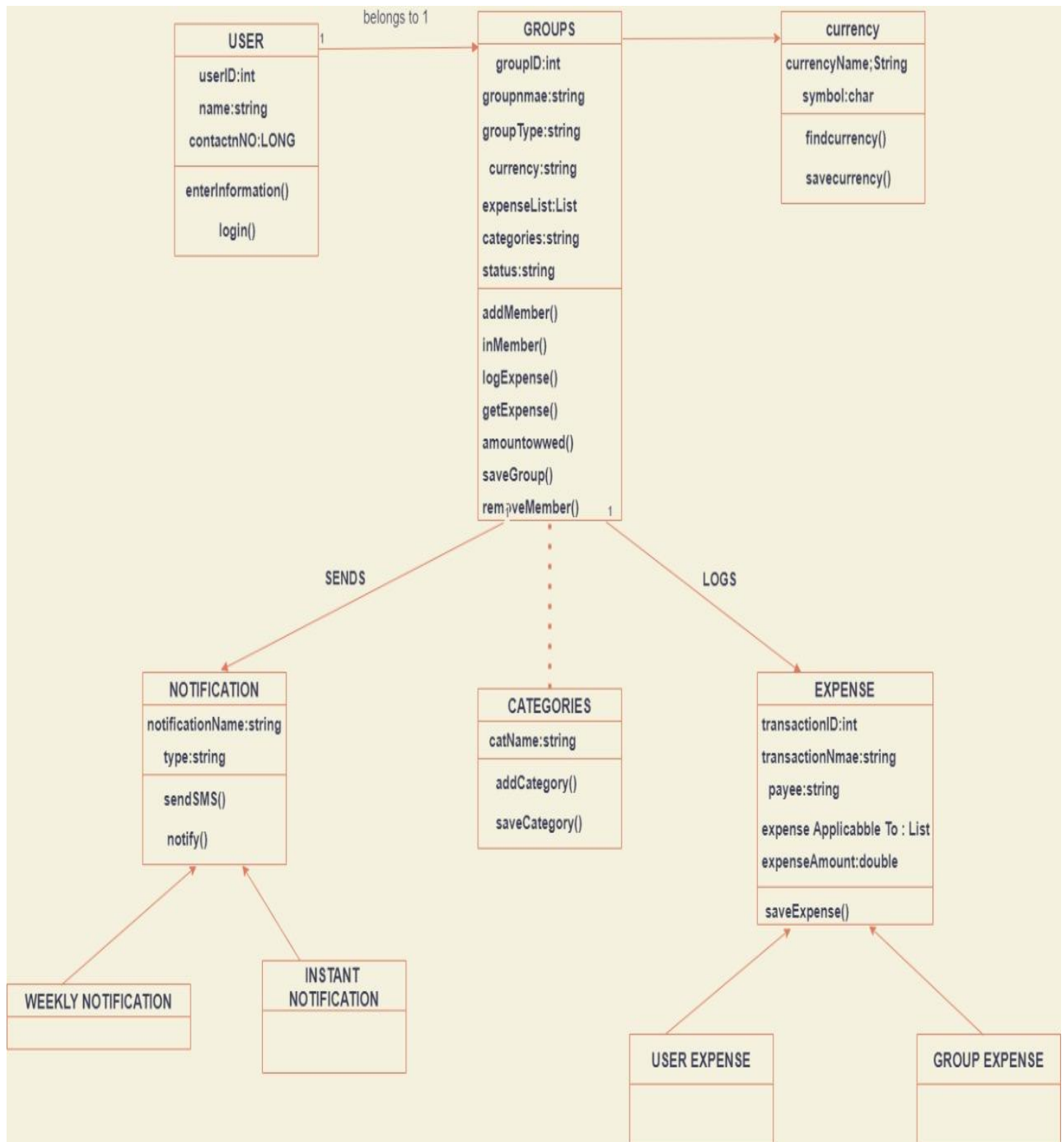
Chapter 4

CONCEPTUAL MODEL
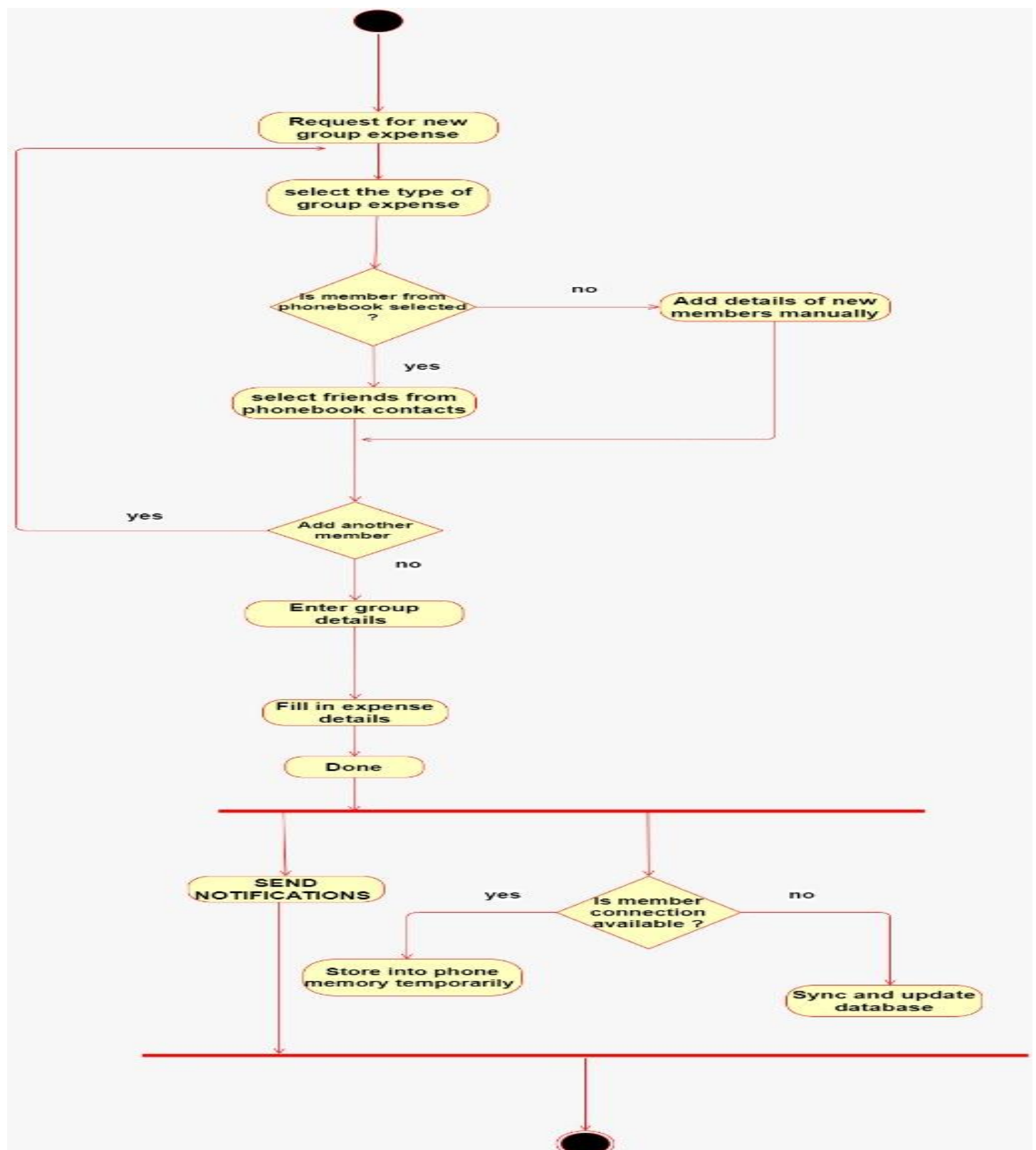
Conceptual Model:

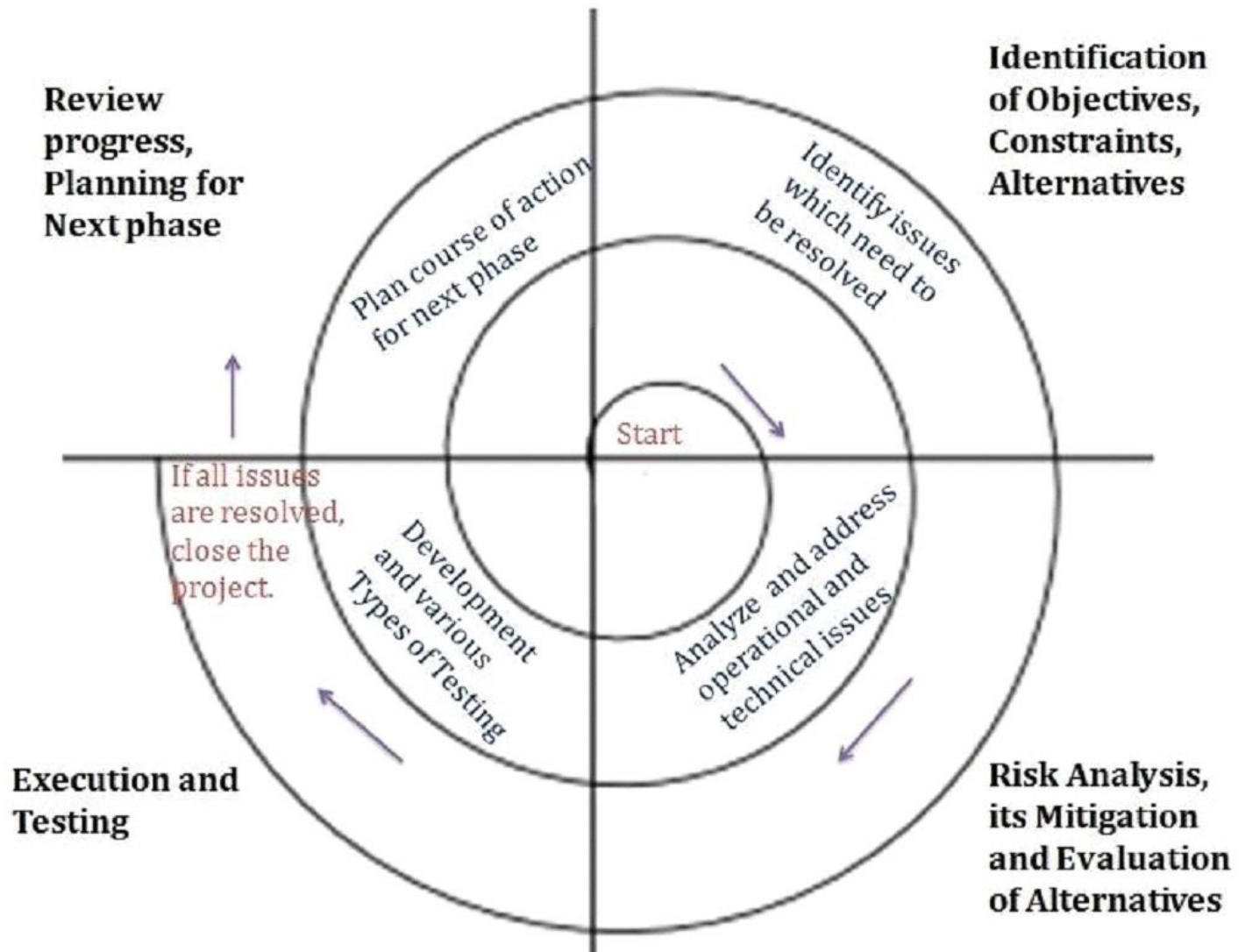## 4.1 Use-case Diagrams:

4.2 E-R Diagrams:

5.3 Class Diagram:

USER

userID:int
name:string
contactnNO:LONG

enterInformation()

login()

belongs to 1

1

GROUPS

groupID:int
groupnmae:string
groupType:string
currency:string
expenseList:List
categories:string
status:string

addMember()
inMember()
logExpense()
getExpense()
amountowwed()
saveGroup()
removeMember()

1

currency

currencyName;String
symbol:char

findcurrency()

savecurrency()

SENDS

LOGS

NOTIFICATION

notificationName:string
type:string

sendSMS()

notify()

CATEGORIES

catName:string

addCategory()

saveCategory()

EXPENSE

transactionID:int
transactionNmae:string
payee:string

expense Applicabble To : List
expenseAmount:double

saveExpense()

WEEKLY NOTIFICATION

INSTANT NOTIFICATION

USER EXPENSE

GROUP EXPENSE

5.4 Activity Diagram:

Chapter 5

<u>Spiral Model</u>

Identification of Objectives, Constraints, Alternatives

Identify issues which need to be resolved

Plan course of action for next phase

Review progress, Planning for Next phase

Start

If all issues are resolved, close the project.

Development and various Types of Testing

Analyze and address operational and technical issues

Execution and Testing

Risk Analysis, its Mitigation and Evaluation of Alternatives

Spiral model is one of the most important Software Development Life Cycle models, which provides support for Risk Handling. In its diagrammatic representation, it looks like a spiral with many loops. The exact number of loops of the spiral is unknown and can vary from project to project. Each loop of the spiral is called a Phase of the software development process. The exact number of phases needed to develop the product can be varied by the project manager depending upon the project risks. As the project manager dynamically determines the number of phases, so the project manager has an important role to develop a product using the spiral model.

The Radius of the spiral at any point represents the expenses(cost) of the project so far, and the angular dimension represents the progress made so far in the current phase.

Each phase of the Spiral Model is divided into four quadrants as shown in the above figure. The functions of these four quadrants are discussed below-

- Objective's determination and identify alternative solutions: Requirements are gathered from the customers and the objectives are identified, elaborated, and analysed at the start of every phase. Then alternative solutions possible for the phase are proposed in this quadrant.
- Identify and resolve Risks: During the second quadrant, all the possible solutions are evaluated to select the best possible solution. Then the risks associated with that solution are

identified and the risks are resolved using the best possible strategy. At the end of this quadrant, the Prototype is built for the best possible solution.

- Develop next version of the Product: During the third quadrant, the identified features are developed and verified through testing. At the end of the third quadrant, the next version of the software is available.
- Review and plan for the next Phase: In the fourth quadrant, the Customers evaluate the so far developed version of the software. In the end, planning for the next phase is started.

Uses of the spiral model:

As mentioned before, the spiral model is best used in large, expensive and complicated projects. Other uses include:

- Projects in which frequent releases are necessary.
- Projects in which changes may be required at any time.
- Long term projects that are not feasible due to altered economic priorities.
- Medium to high-risk projects.
- Projects in which cost and risk analysis is important.
- Projects that would benefit from the creation of a prototype.
- Projects with unclear or complex requirements.

Risk Handling in Spiral Model:

A risk is any adverse situation that might affect the successful completion of a software project. The most important feature of the spiral model is handling these unknown risks after the project has started. Such risk resolutions are easier done by developing a prototype. The spiral model supports copying up with risks by providing the scope to build a prototype at every phase of the software development.

The Prototyping Model also supports risk handling, but the risks must be identified completely before the start of the development work of the project. But in real life project risk may occur after the development work starts, in that case, we cannot use the Prototyping Model. In each phase of the Spiral Model, the features of the product dated and analysed, and the risks at that point in time are identified and are resolved through prototyping. Thus, this model is much more flexible compared to other SDLC models.

Advantages of Spiral Model:

Below are some advantages of the Spiral Model.

- **Risk Handling:** The projects with many unknown risks that occur as the development proceeds, in that case, Spiral Model is the best development model to follow due to the risk analysis and risk handling at every phase.

- **Good for large projects:** It is recommended to use the Spiral Model in large and complex projects.
- **Flexibility in Requirements:** Change requests in the Requirements at later phase can be incorporated accurately by using this model.
- **Customer Satisfaction:** Customer can see the development of the product at the early phase of the software development and thus, they habituated with the system by using it before completion of the total product.

Disadvantages of Spiral Model:

Below are some main disadvantages of the spiral model.

- **Complex:** The Spiral Model is much more complex than other SDLC models.
- **Expensive:** Spiral Model is not suitable for small projects as it is expensive.
- **Too much dependability on Risk Analysis:** The successful completion of the project is very much dependent on Risk Analysis. Without very highly experienced experts, it is going to be a failure to develop a project using this model.
- **Difficulty in time management:** As the number of phases is unknown at the start of the project, so time estimation is very difficult.

Chapter 6
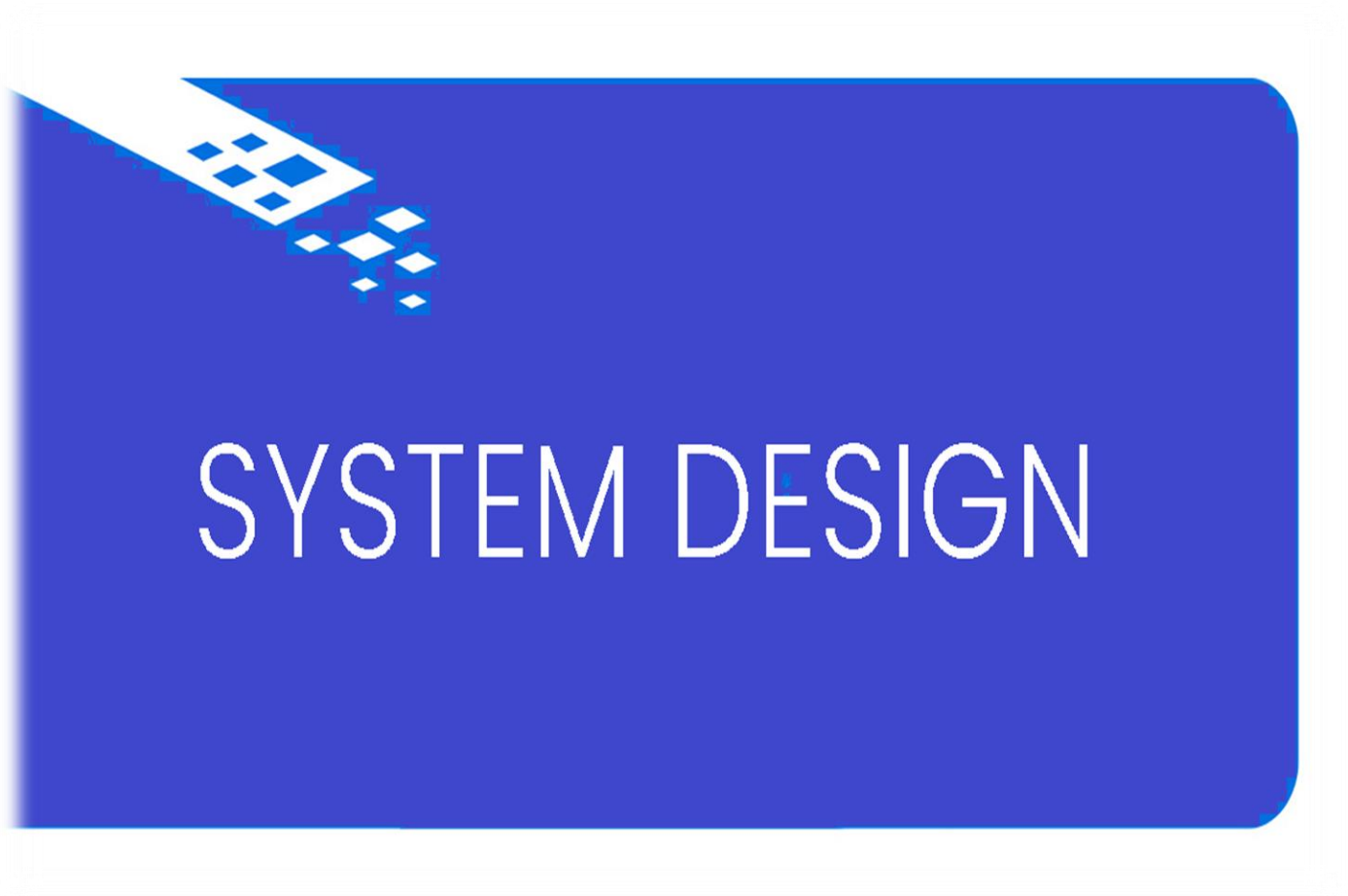
SYSTEM DESIGN

System Design

6.1 Modules in project:

- Add Trip

  Type-in Trip name

- Add Members
  Add group size
  Enter member's name

- Add/Modify Expense
  Select a member → Type-in category →Add/Modify expense

- Expense Graph
  Visualizes the expenses in the form of pie chart

- Calculate Expense
  Divides the expenses of members in equal parts

6.2 Data integrity and constraints

The table shows the constraints applied on the data.

Trip name	Only text allowed
Group size	Only numbers allowed
Member name	Only text allowed
Add category	Only text allowed
Add values (₹)	Only text allowed

6.3 User Interface Design Screenshots:

- ## Start Screen
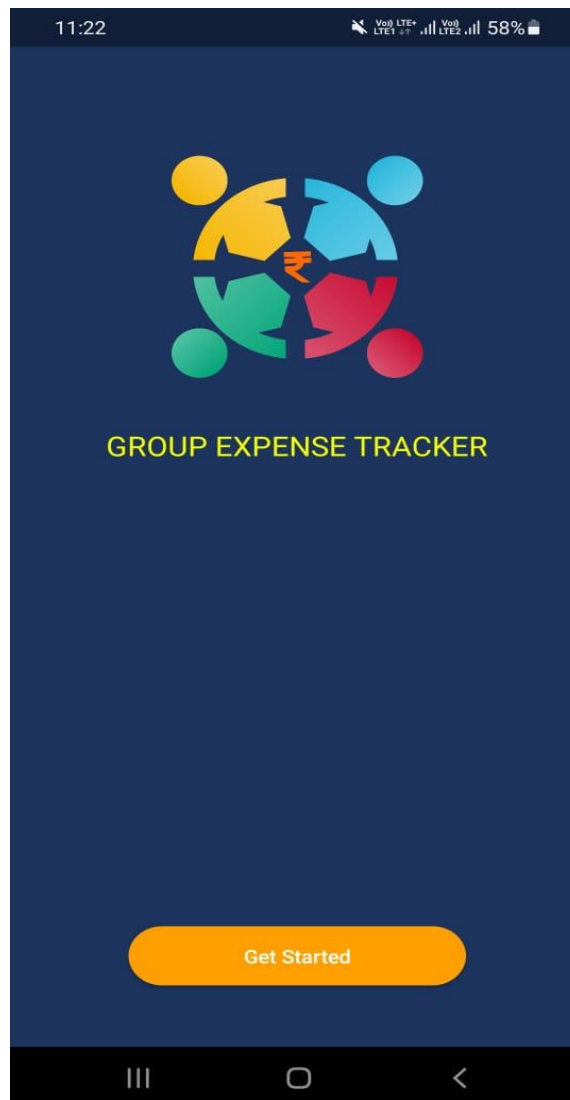
- **Add Trip Name Screen**
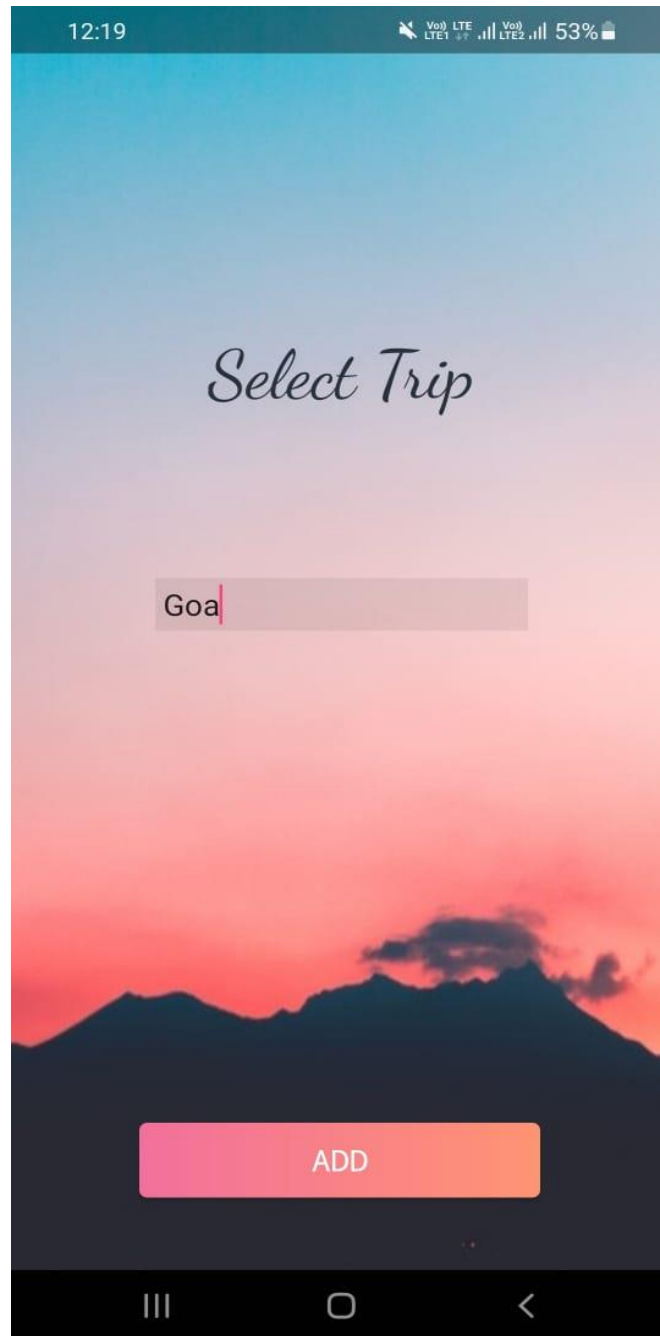
- **Add Group Size Screen**
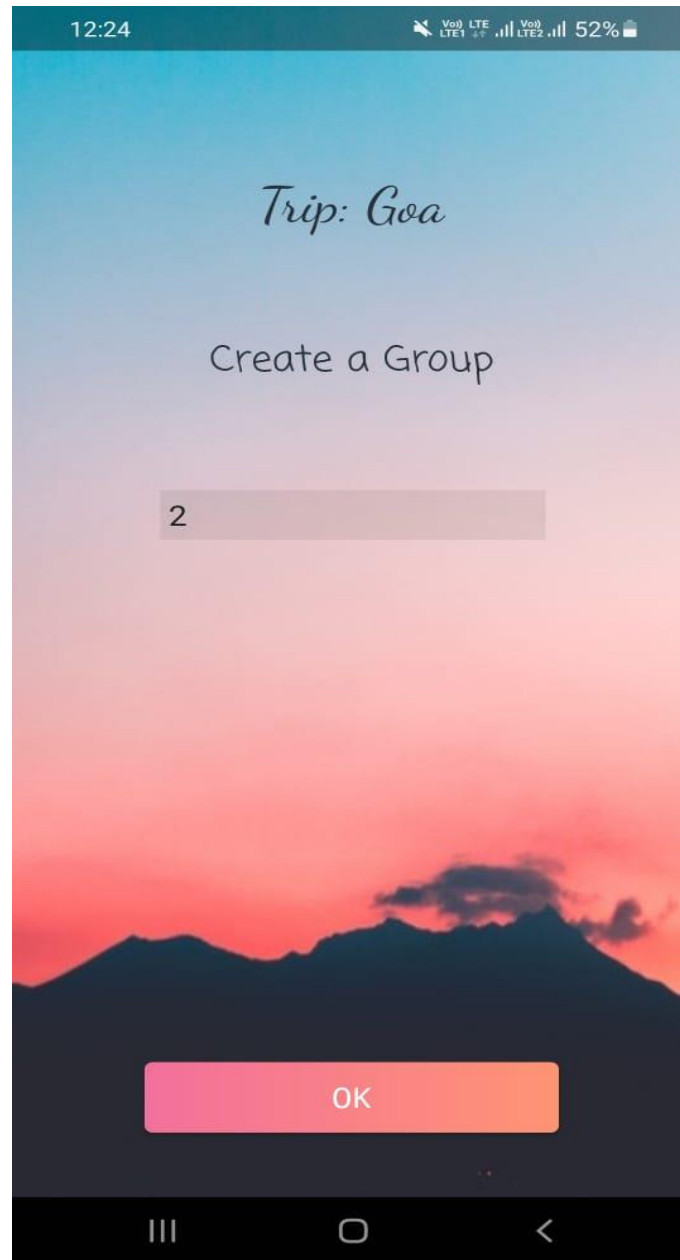
- **Add Group Member's Name Screen**
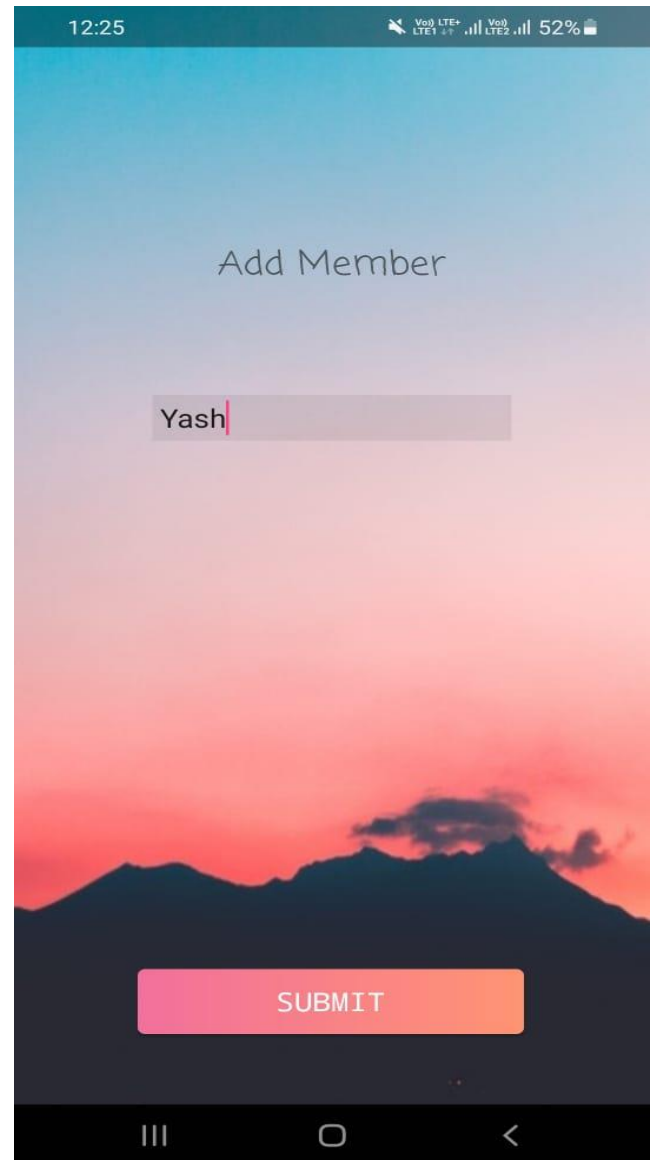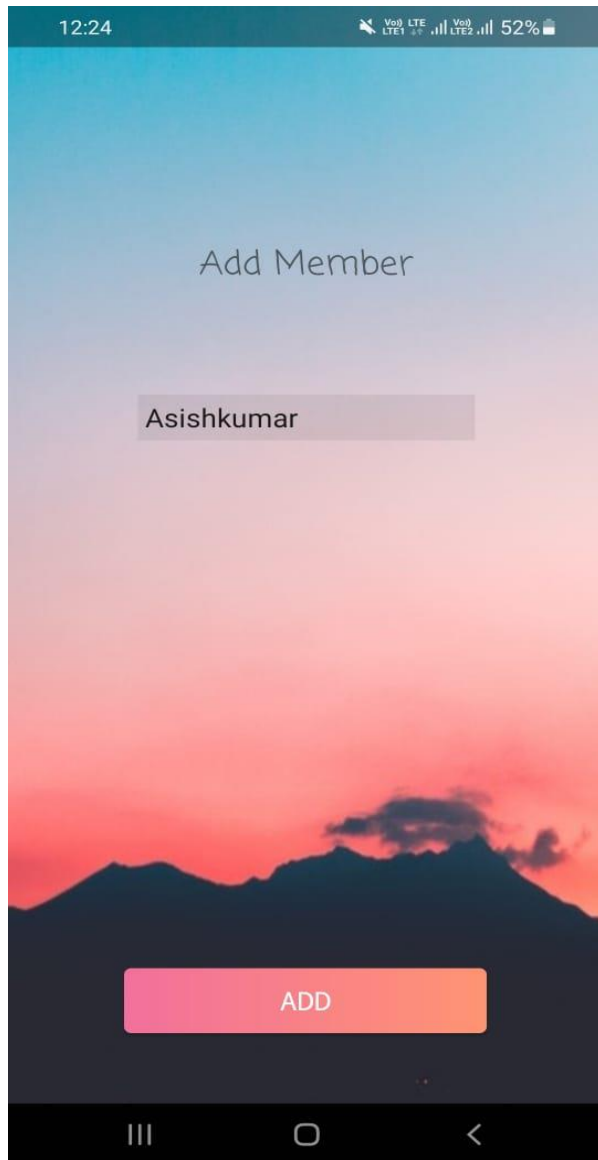
- **Lobby Screen**
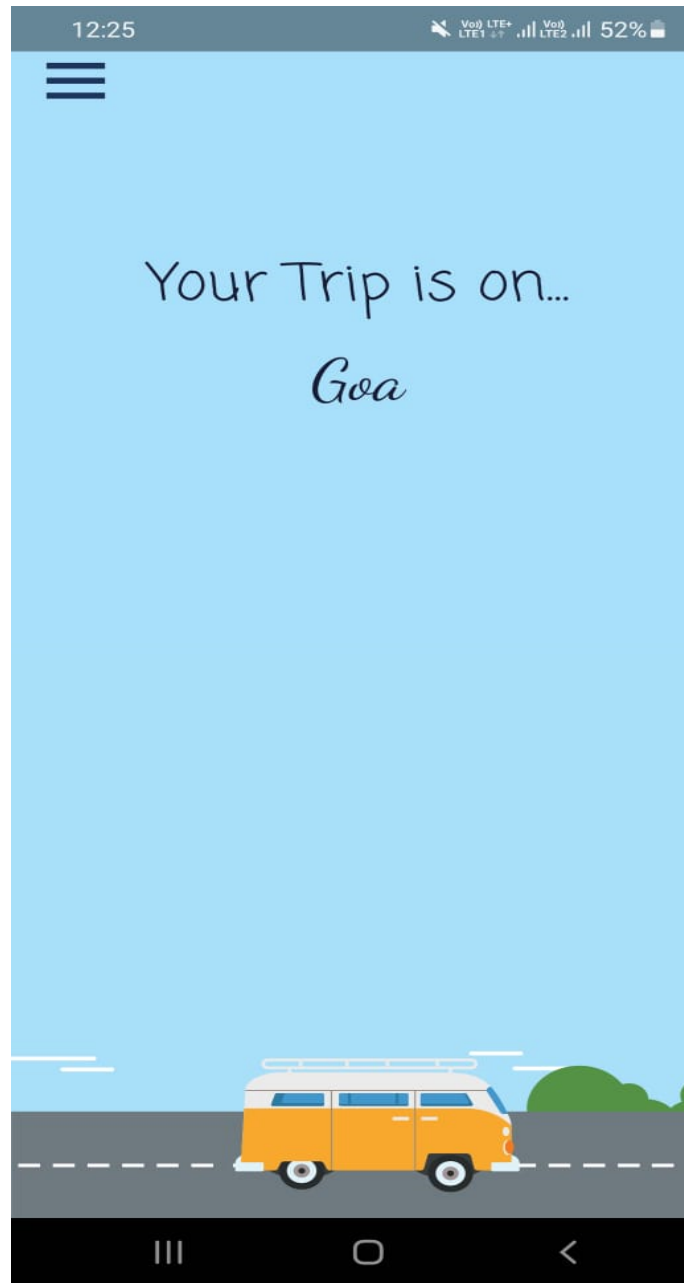
- ## Menu Screen
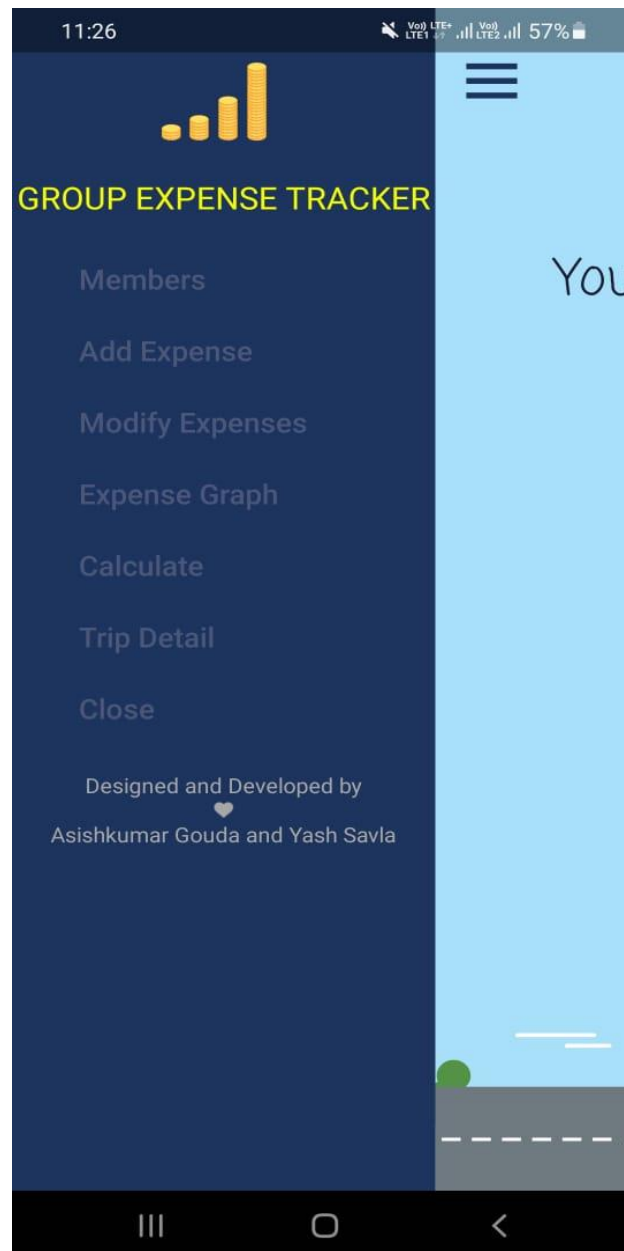
- **Add/Modify Expense Screen**

ADD:

Modify:

- **Expense Graph Screen**
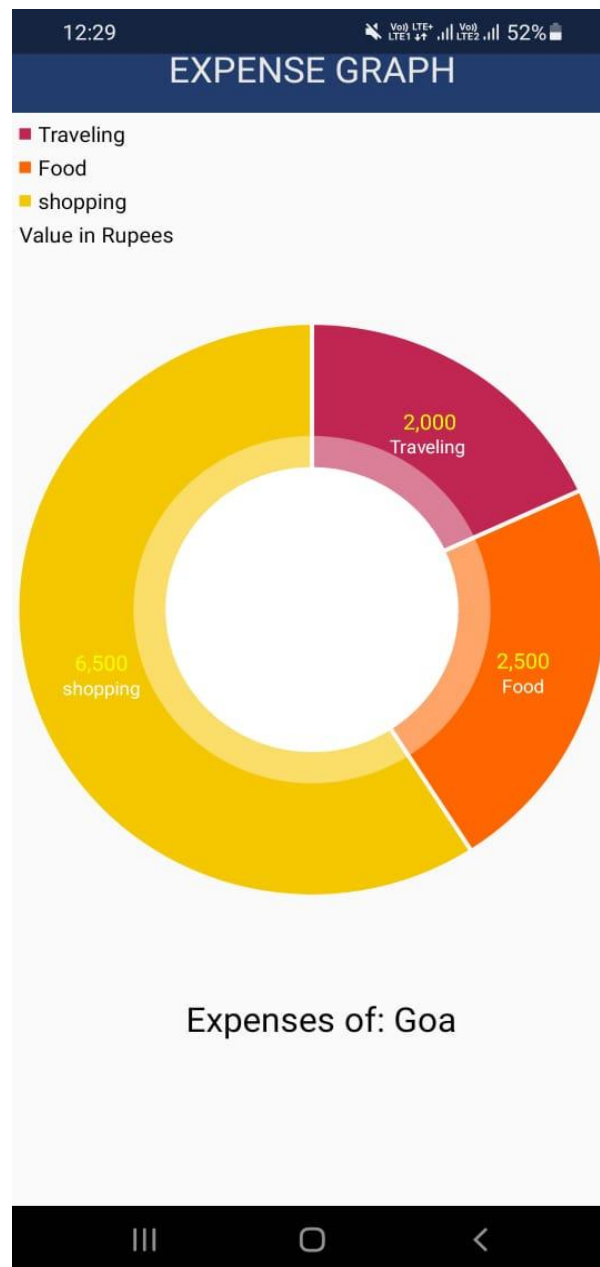
- ## Calculating Expense Screen

- **Trip Detail Screen**

- **Exit Screen**

6.4 Test Case Design

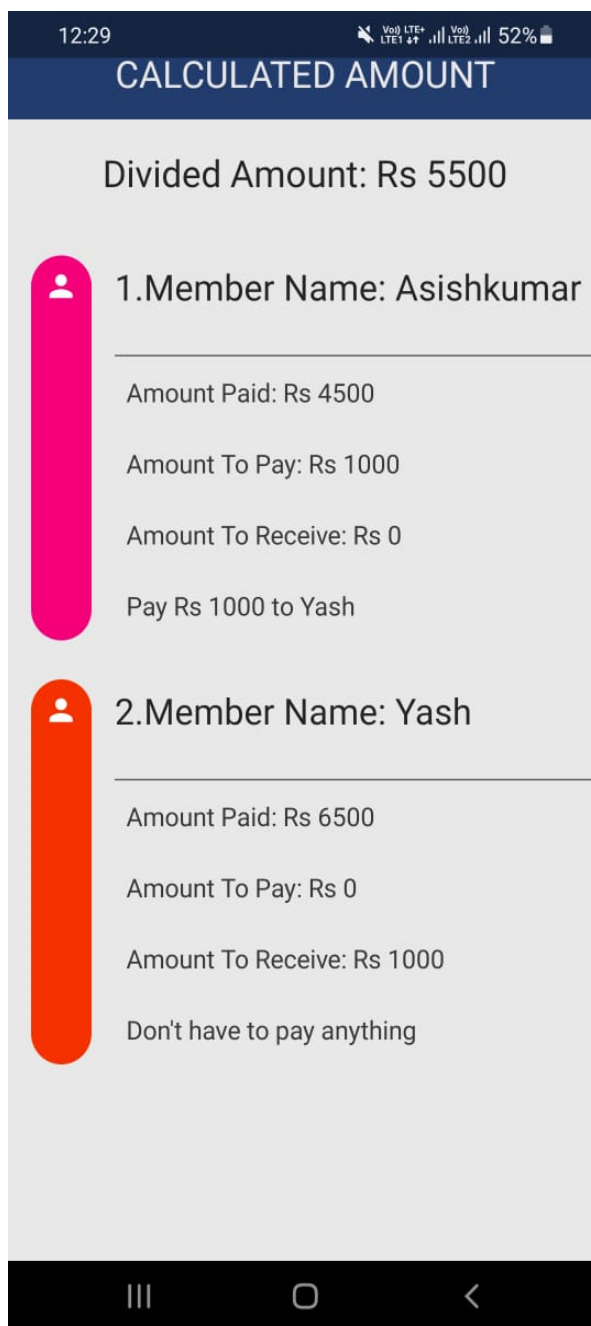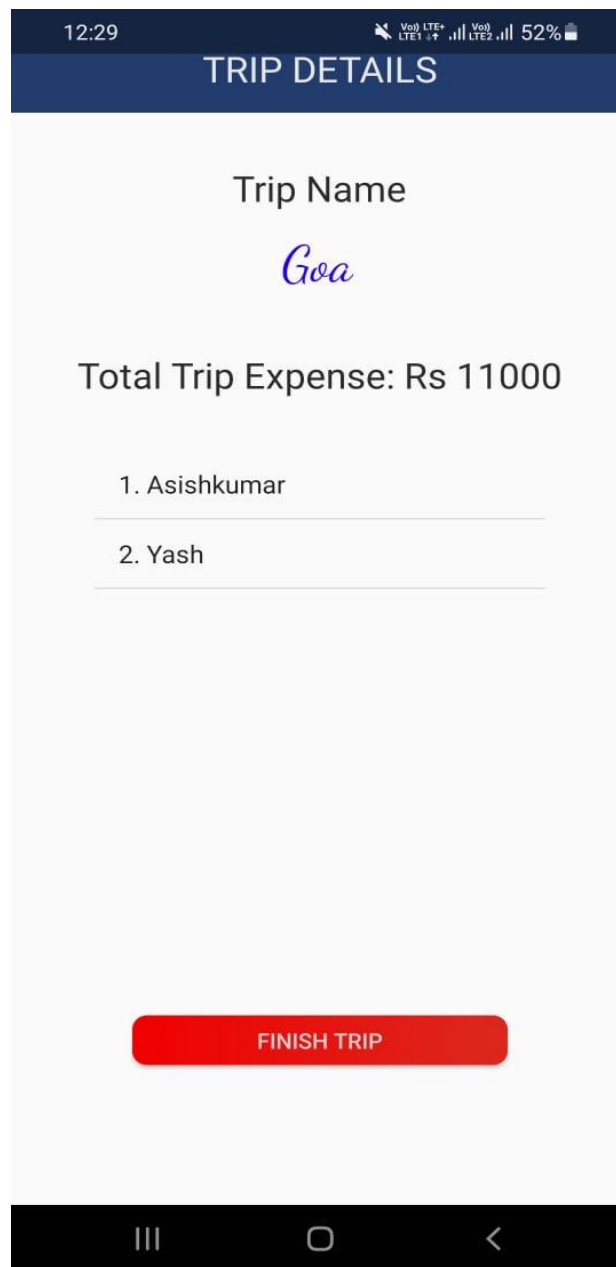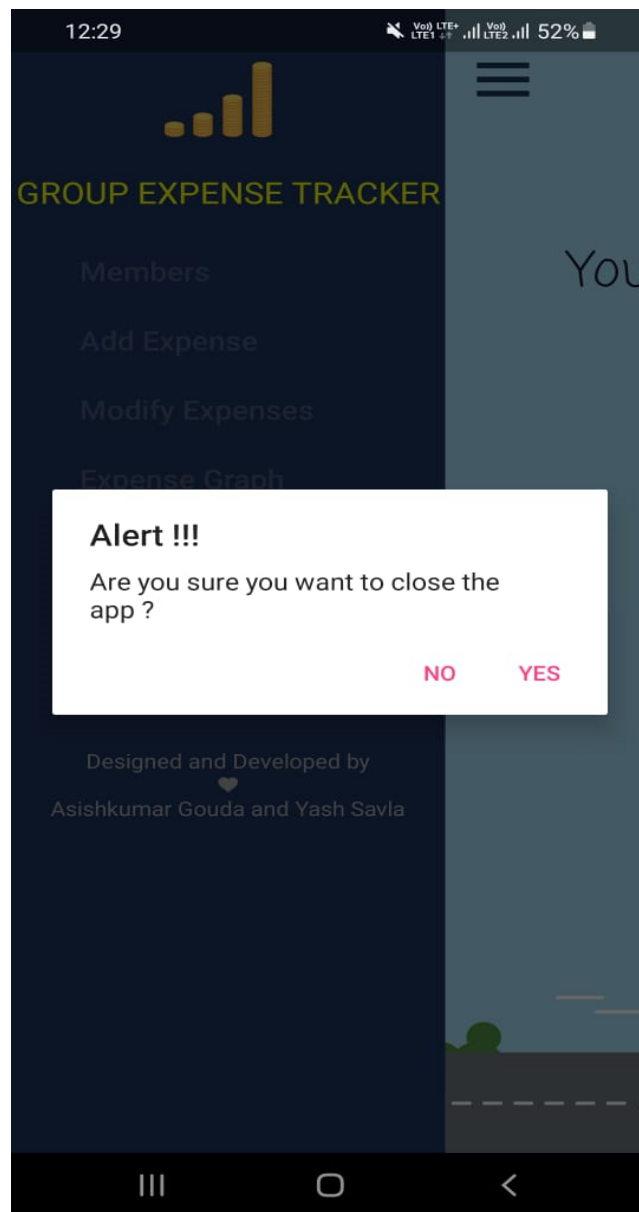Testing is the penultimate step of software development. An elaborate testing of the data is prepared and the system is using the test data. While doing testing, errors are noted and correction is made. Both hardware and software securities are made to run the developed system successfully.

Characteristics of a Good Test:
● Tests are likely to catch bugs
● Not too simple or too complex

Testing
Software testing is a process which is used to measure the quality of software developed. It is also a process of uncovering errors in a program and makes it a feasible task. It is useful process of executing program with the intent of finding bugs. In order to prove that a piece of software works, the software must be tested to determine if the requirements of the application are met. There are several different types of tests used throughout the development process. The two main types of testing are white box and black box testing. White box test cases are used to test specific paths through the code. At decision points you can test the boundaries of the decision (boundary testing) and the partitions of the decision (partition testing).

TYPES OF TESTING

- WHITE-BOX TESTING
- BLACK-BOX TESTING

- **WHITE-BOX TESTING**
  In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs.

- **BLACK-BOX TESTING**
  Black box testing has little or no regard to the internal logical structure of the system, it only examines the fundamental aspect of the system. It makes sure that input is properly accepted and output is correctly produced.

Chapter 7

Implementation and Testing

7.1 FUNCTIONAL TESTING

Functional tests involve exercising the code with nominal input values which gives the expected results and boundary values are known.

7.2 PERFORMANCE TESTING

Performance tests are designed to verify response time. If the wrong data is entered then the system does not allow it and calculations are not performed.

7.3 INTEGRATION TESTING

Integration testing is critical to ensure the functional correctness of the integrated system. Integration testing can be divided into two categories. Integration testing is often the most time consuming and expensive part of testing.

7.4 UNIT TESTING

The first test in the development process is the unit test. The source code is normally divided into modules, which in turn are divided into smaller units called units. These units have specific behaviour. The test done on these units of code is called unit test. Unit test depends upon the language on which the project is developed. Unit tests ensure that each unique path of the project performs

accurately to the documented specifications and contains clearly defined inputs and expected results.

- **Adding credentials Test case**

Sr no.	Input values	Test case	Condition being checked
1	Trip name	Empty	Please enter a valid name of trip
2	Group Size	Empty	Please enter a valid group size
3	Member name	Empty	Please enter a valid name
4	Add expense	Empty	Please enter a valid data
5	Modify expense	Empty	Please enter a valid value

7.5 SYSTEM TESTING

Several modules constitute a project. If the project is long-term project, several developers write the modules. Once all the modules are integrated, several errors may arise. The testing done at this stage is called system test. System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

Chapter 8

RESULTS AND DISCUSSIONS

Result and Discussion

Advantages:

- Builds Discipline and Organization.
- Forces You to Think About Money.
- Crisis Prevention.
- Budget Planner.
- Knowing your spending habits.

Disadvantages:

- Accuracy Issues: A computerized system alone does not ensure accuracy, and the warehouse data is only as good as the data entry that created it.
- The control of inventories is complex because of the many functions it performs. It should be viewed as shared responsibilities.
- The system is not fully automated, it needs data from user for full diagnosis.

Chapter 9

Conclusion

<u>Conclusion</u>

Limitations of the system:

- Credentials are not been saved till all the credentials are being entered in the application.

Future Scope:

- Currently Group expense tracker application doesn't have history feature so it would be added
- Many more currency and its conversion into other currency feature would also be added.

References

- *https://www.sqlite.org/quickstart.html*

- *https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/xmldoc/*

- *https://docs.oracle.com/en/java/*

- *https://developer.android.com/guide*

- *https://github.com/*

- *https://www.youtube.com/*

- *https://www.google.com/*