

LET'S GO

CS 16L2 Mini Project

12130873 Anagha Krishna Nambiar

12130885 Hebah Ashraf

12130893 Joseph Sara Julian

12130917 Sruthi Prasad

B. Tech Computer Science & Engineering



Department of Computer Engineering
Model Engineering College, Thrikkakara
Kochi 682021

Phone: +91.484.2575370

<http://www.mec.ac.in>

hodcs@mec.ac.in

MARCH 2015

Model Engineering College, Thrikkakara
Dept. of Computer Engineering



C E R T I F I C A T E

This is to certify that, this report titled *LETS GO* is a bonafide record of the work done by

12130873 Anagha Krishna Nambiar

12130885 Hebah Ashraf

12130893 Joseph Sara Julian

12130917 Sruthi Prasad

Sixth Semester B. Tech Computer Science & Engineering

students, for the course work in **CS 16L2 Mini Project**, under our guidance and supervision, in partial fulfilment of the requirements for the award of the degree, B. Tech Computer Science and Engineering of **Cochin University of Science & Technology**.

Guide

Reshma Shiekh
Asst. Professor
Computer Engineering

Coordinator

Bijumon T
Asst. Professor
Computer Engineering

Head of the Department

24-3-2015

Ahammed Siraj K K
Associate Professor
Computer Engineering

Acknowledgments

We are extremely thankful to our Principal Prof. Dr. V. P. Devasia for providing us with a conducive environment and the requisite lab facilities. We wish to express our sincere thanks to Mr. Ahammed Siraj K K, Head of the Department, Department of Computer Science and Engineering, for his inspiration and constant encouragement which made us take up the project and bring it to the completion. We are indebted to our project coordinator Mrs. Manju K, Assistant Professor, Department of Computer Science and Engineering, for her constant help and support. We extend our sincere and heartfelt thanks to our guide Mrs. Anne Jacob Silja for helping us throughout the course of this project by providing us with valuable advice. We thank God Almighty for all the blessings showered on us during this endeavor. Not but not the least we thank our colleagues for the support and encouragement they have given us during the course of our work.

Anagha Krishna Nambiar
Hebah Ashraf
Joseph Sara Julian
Sruthi Prasad

Abstract

Through this android application, we aim to help students, who often find it difficult to travel home alone in the first year of college, identify other students who are traveling to the same place. This can be a great way to make new friends too! This app maintains a database in the cloud with the details such as date of travel and destination of all the users that sign in and then it will detect other users with common entries. Every member of this selected group of users will receive push notifications regarding their co-passengers. Lets Go helps you plan your journeys easily and effectively.

Contents

1	Introduction	1
2	Literature Survey	2
2.1	Existing Systems	2
2.1.1	Limitations of Existing Systems	2
2.2	Proposed System	2
2.2.1	Advantages of Proposed System	2
3	Proposed System	3
3.1	Problem Statement	3
3.2	Proposed Solution	3
4	Software Requirement Specification	4
4.1	Introduction	4
4.1.1	Purpose	4
4.1.2	Intended audience	4
4.1.3	Project Scope	4
4.1.4	Design and Implementation Constraints	4
4.1.5	Assumptions and Dependencies	5
4.2	Functional and Non-Functional Requirements	5
4.2.1	Functional Requirements	5
4.2.2	Create account	5
4.2.3	Update Status	5
4.2.4	Matching destination	5
4.2.5	Privacy	5
4.2.6	Co-passenger details	5
4.2.7	Notify	5
4.2.8	External Interface Requirements	6
4.2.9	Non Functional Requirements	6
4.2.10	Business Rules	6
4.3	Hardware & Software Requirements	6
4.3.1	Hardware Requirements	6
4.3.2	Software Requirements	7

Short project name	Contents
5 System Design	8
5.0.3 Architectural Design	9
5.0.4 Design Rationale	10
5.1 Data Design	10
5.1.1 Entities and Relationships	10
5.1.2 Data Description	10
5.1.3 Data Dictionary	10
5.2 Algorithms	10
5.3 Human Interface Design	11
5.3.1 Overview of User Interface	11
5.4 Requirements Matrix	12
5.5 Dataflow Diagrams	13
5.6 Usecase Diagrams	14
5.7 Class Diagrams	15
5.7.1 ER Diagram	15
6 Implementation	16
6.1 Overview of Technologies Used	16
6.1.1 Android	16
6.1.2 XML	16
6.1.3 Parse	16
6.2 Testing	16
6.2.1 Types of Testing	16
6.3 Results	17
7 Conclusion	19
7.1 Conclusion	19
7.2 Future Scope	19
References	20

Chapter 1

Introduction

As freshers in college, we too faced similar situations and had no where to turn to. We believe empowering students with such an application in their first year of college itself would help solve the difficulty of finding co-passengers on their way home. Lets Go was developed on the android platform with the intention of reaching out to most of the mobile users. The application maintains a database in the cloud with details such as, date of travel and destination of all the users that sign in. Using this information, the app can form groups based on common hometowns. Every member of this group will then receive push notifications regarding their co-passengers. Lets Go helps you reach home safely with friends.

Chapter 2

Literature Survey

2.1 Existing Systems

There are several android applications that provide travel advisory.

2.1.1 Limitations of Existing Systems

- Most of the apps do not help find co-passengers.
- There are no application that focuses on the journey home.

2.2 Proposed System

Students face the difficulty of finding co-passengers during their journey to home at weekends and holidays. Users of this application can update a status "going to" mentioning their destination and date of travel. It then, forms a group of users who have a common destination and same travel date and send push notifications to all members of the group.

2.2.1 Advantages of Proposed System

- Ease of finding co-travellers.
- Helps make friends in the first year.
- It focuses on every trip to home.

Chapter 3

Proposed System

3.1 Problem Statement

Travel is a quintessential part of our lives. Moreover, the farther the destination is, more tedious the travel is. In College, every students face the difficulty of finding co-passengers during their journey to home at weekends and holidays. Such situations require a common platform for a group of users so that people travelling to the same place can connect easily and plan their trip together.

3.2 Proposed Solution

What we aim to create is an application that takes input from multiple users, groups them based on their destination and date of travel, which is recorded in the cloud database. The members of this group will be notified and informed about other co-passengers and also asked to confirm their travel before list of details of co-passengers is displayed.

Input: Users of this application can update a status "going to" mentioning their destination and date of travel.

Output: Forms a group of users who have a common destination and same travel date and send push notifications to all members of the group.

Chapter 4

Software Requirement Specification

4.1 Introduction

4.1.1 Purpose

The purpose of the project is to make the journey of hosteler students to their homes, easier. This application connects students with common hometowns(target destinations of journey) and common date of journey and helps them to travel together with people whom they would like to travel with.

4.1.2 Intended audience

The intended audience for the project are the whole of the Android users. This document is intended to inform the reader with the approach to create the LET'S GO application. The reader will get a general understanding of the product including its functional, non- functional requirements and its features. The document begins with the introduction of the application, followed by general features of the application, target users, design consrains, and interface requirements. All these details are specified in detail. Reading through these sections clearly gives an idea of the various challenges in the development of this project as well as the standards and guidelines that are to be adhered to.

4.1.3 Project Scope

The project will be developed using Android Studio, as an Android application which can be used comfortably by the targeted users.The target users include hosteler students.They can make use of this application for going home during weekends and during holidays if they don't have a companion.The users should have this Android application installed in their smartphones.

4.1.4 Design and Implementation Constraints

This project is to be completed, including coding, testing and loading of the database, by March 31, 2015.

4.1.5 Assumptions and Dependencies

No future factors could affect the requirements stated in the SRS, as this application will be developed for a limited audience.

4.2 Functional and Non-Functional Requirements

4.2.1 Functional Requirements

1. Create account.
2. Update Status
3. Matching Destination
4. Privacy
5. Co-passenger details
6. Notify

4.2.2 Create account

The user can enter the application by creating a new account by giving the necessary details.

4.2.3 Update Status

The user should be able to update his/her going to status for their trip.

4.2.4 Matching destination

The users update their 'going to' status giving details about the starting point , travel date and destination.If the status matches, the user have the options to decide their co-passengers.

4.2.5 Privacy

The user can set their privacy. The application provides options for the users to set their privacy according to their needs. The user can decide who all can see their statuses.

4.2.6 Co-passenger details

Once the user has selected their co-passengers, he or she will be provided with the details of the people who are going to travel with them.

4.2.7 Notify

The user must be notified of the co-passenger details even when he/she is logged out of this.

4.2.8 External Interface Requirements

4.2.8.1 User Interfaces

A touchscreen android device is used as the interface.

4.2.8.2 Software Interfaces

Parse APIs.

4.2.8.3 Communication Interfaces

Parse Backend services are used.

4.2.9 Non Functional Requirements

4.2.9.1 Performance Requirements

Capacity requirements: Capacity to be used by several users, especially in the start of a holiday season, approximately 500 to 1000 users. Database will have to include 5(name, phone no, hometown, destination, college name) particular informations about each user. Response-time: 512 kbps

4.2.9.2 Safety Requirements

User will have to provide his/her phone number and destination which she can share with other users only if the user gives her consent to. The application will not leak out any information that is not public already. Just like files saved on the device's internal storage, Android stores the database in private disk space that's associated with the application. This data is secure, because by default this area is not accessible to other applications.

4.2.9.3 Software Quality Attributes

The application can be made available to several users from different colleges, yet going to the same place. Users will find it adaptable to their specific locations and positions. Developers, hence have to keep in mind that the application will be flexible and has to be maintained. The application is going to be made for android phones, which rules out any future hopes of it being portable to other environments. It will be very easy to use for the targeted audience.

4.2.10 Business Rules

Since this product mainly focuses on the freshers in college, they can use this app, as intended, when they wish to travel with friends, while going home. Under other circumstances, they can use the same app for travelling to other places too.

4.3 Hardware & Software Requirements

4.3.1 Hardware Requirements

- Mobile phone

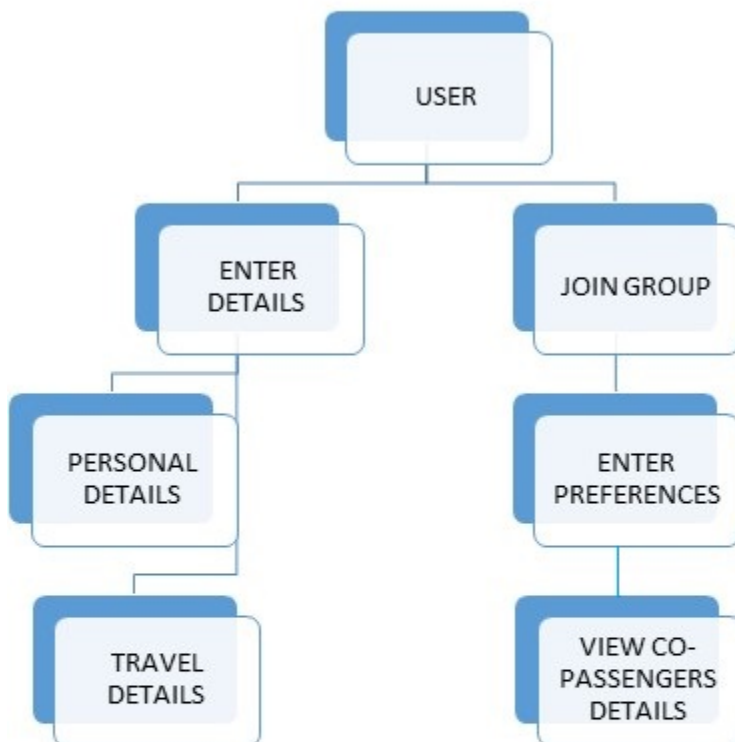
4.3.2 Software Requirements

- Android Studio
- Parse

Chapter 5

System Design

5.0.3 Architectural Design



5.0.4 Design Rationale

The above architecture is best suited for an android application which stores the travel details of one user at a time.

5.1 Data Design

5.1.1 Entities and Relationships

The various entities used in the product are:

1. Student
2. Place

5.1.2 Data Description

Data is stored in text files stored in the Parse server.

5.1.3 Data Dictionary

1. Place

Destination	varchar(20)	primary key	Destination
Starting_point	varchar(8)	not null	Starting point
Travel_date	date	not null	Travel date

2. Student

college	varchar(10)	not null	College
destination	varchar(40)	primary key	Destination
ph_no	int(20)	not null	Phone number
St_name	varchar(8)	primary key	Student name
Tr_date	date	not null	Travel date

5.2 Algorithms

CreateAccount Algorithm create

Input: User's personal information such as Name, College, Hometown, Phone no, User name, Password.

Output: A new entry in database.

Procedure:

Tabulate Entries entered by the user in the respective columns

sqlName=getName()

sqlGender=getGender()

sqlCollege=getCollege()

sqlHometown=getHometown()

sqlPhone=getPhone()

sqlUsername=getUsername()

sqlPassword=getPassword()

Login()

Input:Username, password

Output: Entry to view past trip details/and or update new trip details

Procedure:The Username and password are matched with those recorded in the database already, else a prompt to sign up is shown.

Goingto()

Input: Starting point, Destination, travel date and time.

Output: Trip details will be recorded. A notification may be sent with co-passenger details

Procedure: Enter the required details in the proper format and record it in the database.

Approve()

Input:The User will be asked to select their preferences of the co-passengers via tick buttons stored as choice

Output: The selection will be recorded as 'choice'

Procedure:choice=Getchoice()

ViewGroup()

Input:Selection from Approve()

Output: Chosen co-passegners will be shown

Procedure:Select * Student where Gender is Choice

Exitapp()

Input: The users closes the app by pressing the back button.

Output: A prompt will be given to the user

Procedure:If user chooses to exit saving call savedetails() which will save trip details just entered else

exit() without saving any prior settings and information.

Quitgroup()

Input:The user choose to quit the group she/he was put into in case of expiry of travel date or any such situation.

Output: All the group details will be deleted from the database.

Procedure: DELETE * FROM Student,Place

5.3 Human Interface Design

5.3.1 Overview of User Interface

The Let's Go android application enables the user to create an account by providing the basic user details.The user waits until a push notification regarding the co-passenger details are recieved. With this notification user can accept/reject the group invitation depending on the safety requirements.Later, after entering a group with the satisfied co-passengers the trip details can be planned accordingly.

5.4 Requirements Matrix

	createAccount	Login	Goingto	FormGroup	Notify	Approve	ViewGroup	Quitgroup
CreateAccount	1	0	0	0	0	0	0	0
UpdatingStatus	0	1	1	0	0	0	0	0
Privacy	0	1	0	0	0	1	1	0
Matching Destination	0	1	1	1	0	0	0	0
Copassenger Details	0	1	0	0	0	1	1	0
Notify	0	1	1	1	1	0	0	0
Save Details	0	1	1	1	1	1	1	0
Quit Group	0	1	0	1	0	0	0	1

5.5 Dataflow Diagrams

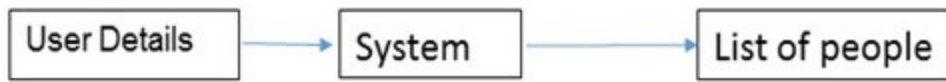


Fig2. Level 0

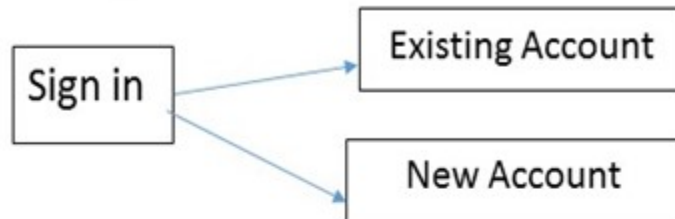


Fig 3. Level 1

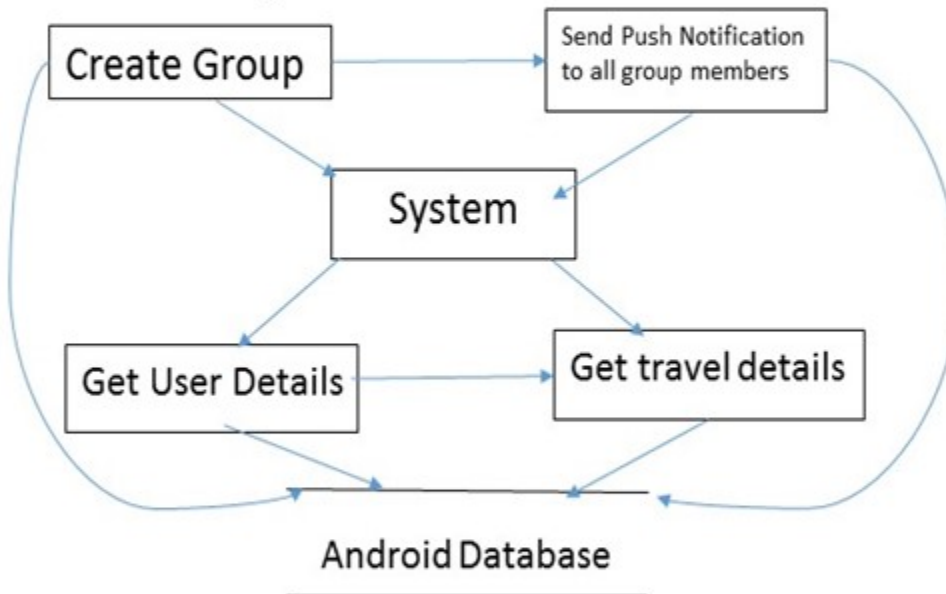
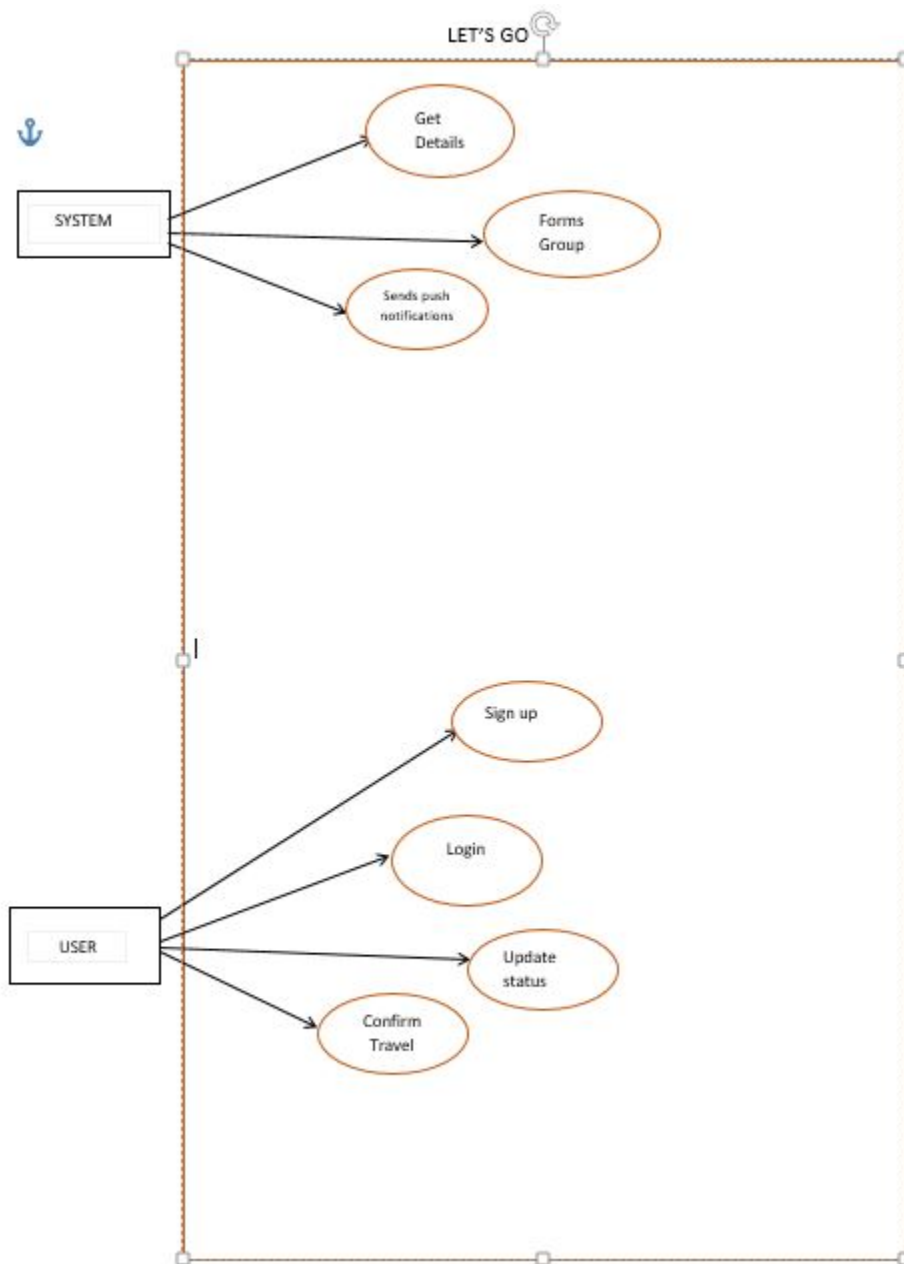
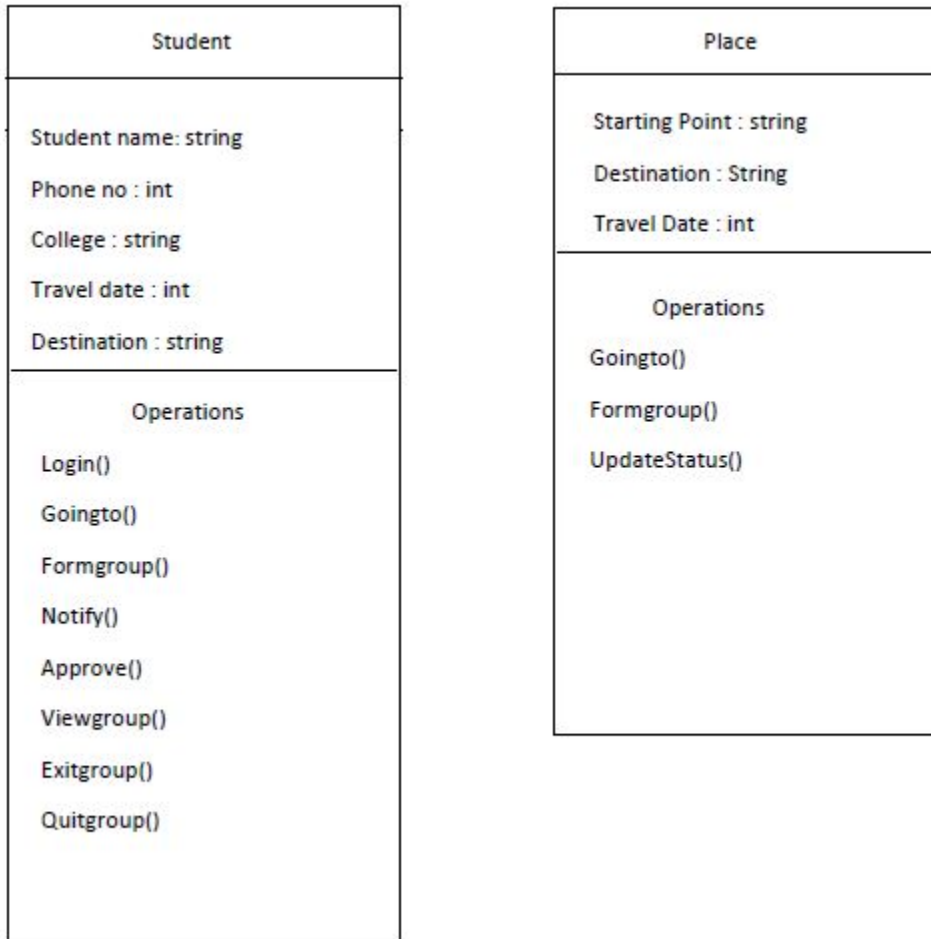


Fig 4. Level 2

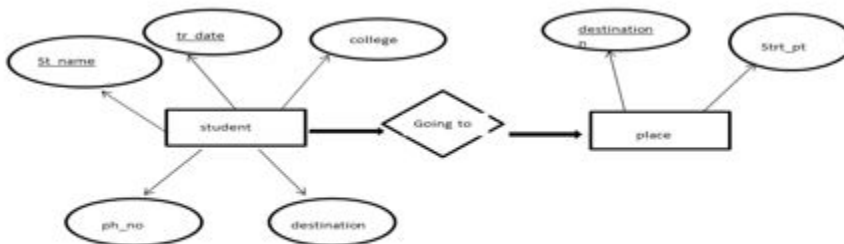
5.6 Usecase Diagrams



5.7 Class Diagrams



5.7.1 ER Diagram



Chapter 6

Implementation

6.1 Overview of Technologies Used

6.1.1 Android

Android Programming through Java forms the base of this system. Majority of the algorithmic functions are done through Java.

6.1.2 XML

XML forms the base for the UI design of this model.

6.1.3 Parse

Parse forms the back-end for the application. It is a model for providing Web and mobile app developers with a way to link their applications to back-end cloud storage, while also providing features such as user management, push notifications and integration with social networking services.

6.2 Testing

6.2.1 Types of Testing

6.2.1.1 Alpha Testing

The database was populated with multiple accounts logged in at the same time. Synchronisation of shared lists was checked by performing multiple delete and add operations simultaneously.

6.2.1.2 Beta Testing

The app was distributed to a select group of users to test the application for push notifications and to view the co-passenger details.

6.3 Results

Login

Create Account Existing User

Name :

Gender :

Phone no. :

Home town :

College Name :

Username

password

SUBMIT

Login

Username

Password

Login

OR

Sign Up

Login

From Edappally Toll

Destination Kannur

Current Date:

Choose date 24 Apr 2015

Search for Co-travellers

Login

From Emakulam North

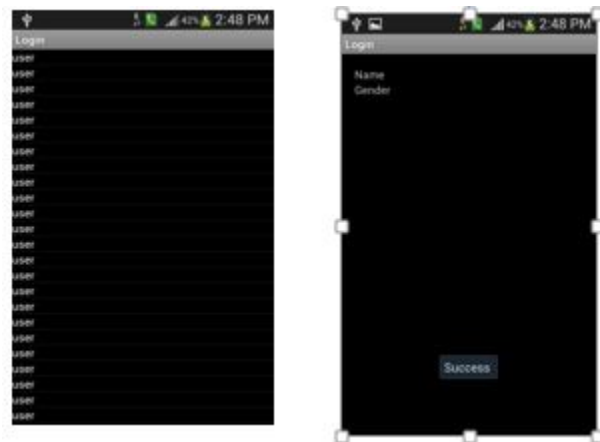
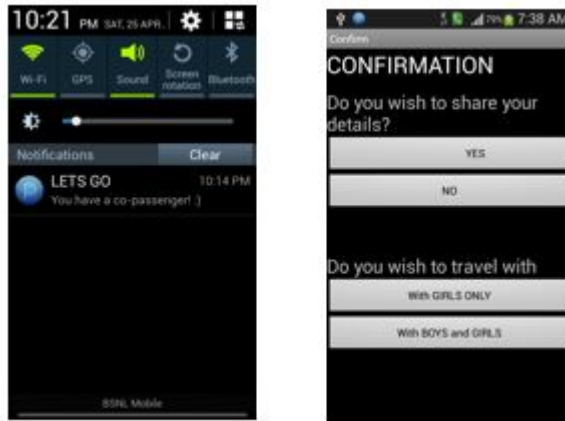
Destination Kannur

Kannur

Pathanamthitta

Cherai

Palakkad



Chapter 7

Conclusion

7.1 Conclusion

Albeit a small step in empowering the students of College with the ease of finding co-travellers, this application paves the way for making the journey home easier and in connecting people going to the same destinations. By eliminating the need to searching out partners for travelling, this application allows its users to make new friends in college.

7.2 Future Scope

1. APIs of Maps can be included to help track the routes of the journey.
2. Groups can be formed by users themselves
3. Chat boxes can be added to help connect people.

References

- [1] “Android Programming)” [Online].
Available: *[http : //www.developer.android.com/](http://www.developer.android.com/)*
- [2] “Android Programming)” [Online].
Available: *[http : //www.udacity.com/courses/android/](http://www.udacity.com/courses/android/)*