

**F1 : The system shall allow the user to access all the fields.**

- **Description :** It is a home page which has Home, About, Reviews, Get Started, and Contact button.
- **Input :** User can click on any button for further information.
- **Processing :** System will allow the user to enter on that particular page with credentials, if required.
- **Output :** User is prompted to the page, he/she clicked.

**F2 : The system shall allow the user to sign in.**

- **Description :** Sign In page for the users .
- **Input :** Users enter their login IDs and password and click on the login button.
- **Processing :** System will authenticate the user with its credentials.
- **Output :** After successful authentication, the system prompts the user to next step what he/she wants .

**F3 : The system shall allow the user to register themselves.**

- **Description :** Register page for users.
- **Input :** Users have to enter the following credentials:- Name, Course, Address, Email, Phone number, Age, Gender, Password, etc.
- **Processing :** Application sends the entered information to the server and server verifies it.
- **Output :** After a successful registration, the user can go to the next step.

**F4 : The system shall allow the user to edit one's Profile.**

- **Description :** If anything the user wants to update, can update through this.
- **Input :** After clicking on the edit button, the system will allow the user to edit the profile.
- **Processing :** Users profile is updated as per his need.
- **Output :** Updates profile with all the changes.

**F5 : The system shall allow the user to create a group.**

- **Description :** Users can create a group for the journey.
- **Input :** Users have to give credentials like Travel Location, Vehicle, Total Member, Total fare, etc.
- **Processing :** System will create a group with those information users gave.

- **Output :** A group is created successfully.

**F6 :** The system shall allow the user to see all the groups which exist.

- **Description :** For displaying the groups that were created already.
- **Input :** Users have to click on the open button which redirects users to the group details .
- **Processing :** System will display the groups which exist.
- **Output :** Displays the existing groups successfully.

**F7 :** The system shall allow the user to display the group details of a particular group.

- **Description :** User can display the group details of a particular group.
- **Input :** Users have to give the name of the admin of that particular group.
- **Processing :** System will display the details, if any group exists.
- **Output :** Details of that group displayed.

**F8 :** The system shall allow the admin to remove groups from the existing groups .

- **Description :** Admin can remove the groups when the journey is completed.
- **Input :** Group name with leaving date is required for removing the group.
- **Processing :** The system will allow the admin to remove the group.
- **Output :** Group is removed successfully.

<Leave this page blank>



# **NATIONAL INSTITUTE OF TECHNOLOGY CALICUT**

## **SOFTWARE REQUIREMENTS SPECIFICATION**

**FOR**

**GROUP NO. : 10**

**PROJECT NAME : NITC TRAVEL TOGETHER**

**Prepared By:**

M210687CA	Karan Sachinkumar Satish	<a href="mailto:karan_m210687ca@nitc.ac.in">karan_m210687ca@nitc.ac.in</a>
M210663CA	Atul Kumar Sethi	<a href="mailto:atul_m210663ca@nitc.ac.in">atul_m210663ca@nitc.ac.in</a>
M210702CA	Shivam Kumar Jha	<a href="mailto:shivam_m210702ca@nitc.ac.in">shivam_m210702ca@nitc.ac.in</a>

**Instructor:**

**ADHIKARI VENKATA YUKTHA**

**COURSE:**

**SOFTWARE ENGINEERING LABORATORY**

**DATE: 24-Jan-2023**

## CONTENTS

<b>1 Introduction.....</b>	<b>8</b>
<b>1.1 Document Purpose.....</b>	<b>8</b>
<b>1.2 Product Scope.....</b>	<b>8</b>
<b>1.3 Intended Audience and Document Overview.....</b>	<b>9</b>
<b>1.4 Definitions, Acronyms and Abbreviations.....</b>	<b>9</b>
<b>1.5 Document Conventions.....</b>	<b>10</b>
<b>1.6 References and Acknowledgments.....</b>	<b>11</b>
<b>2 Overall Description.....</b>	<b>12</b>
<b>2.1 Product Overview.....</b>	<b>12</b>
<b>2.2 Product Functionality.....</b>	<b>13</b>
<b>2.3 Design and Implementation Constraints.....</b>	<b>13</b>
<b>2.4 Assumptions and Dependencies.....</b>	<b>14</b>
<b>3 Specific Requirements.....</b>	<b>15</b>
<b>3.1 External Interface Requirements.....</b>	<b>15</b>
<b>3.1.1 User</b>	
<b>Interfaces.....</b>	<b>15</b>
<b>3.1.2 Hardware Interfaces.....</b>	<b>19</b>
<b>3.1.3 Software Interfaces.....</b>	<b>19</b>
<b>3.2 Functional Requirements.....</b>	<b>19</b>
<b>3.3 Use Case Model.....</b>	<b>22</b>
<b>3.3.1 Use Case #1.....</b>	<b>23</b>

3.3.2 Use Case #2.....	23
.....	
3.3.2 Use Case #6.....	26
4 Other Non-functional Requirements.....	27
4.1 Performance Requirements.....	27
4.2 Safety and Security Requirements.....	27
4.3 Software Quality Attributes.....	27
5 Activity Log.....	28

## Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Riya Solanki, Khushboo Patidar, Amay Wakde	Initial version of the project plan document	24 January,2023
1.1	Riya Solanki, Khushboo Patidar, Amay Wakde	Modified version of the Software Requirement Specification, following changes has been done - Diagram labelling,  Extra information has been removed and input and output fields have been added.	28 January,2023
2.0	Karan Sachinkumar Satish,Atul kumar Sethi,Shivam Jha Karan	This is essentially a Design Document. UML diagrams including Class diagrams, Object diagram, Sequence diagram, Activity Diagram is added.	07 February,2023
2.1	Sachinkumar Satish,Atul kumar Sethi,Shivam Jha	Modified Version of the design document. All the suggested changes made in ER diagram, Activity diagram, Testcases added and references added.	10 February,2023
2.2	Subhajit Majumdar Prakash Singh Debayan Ghosh	Modified ER Diagram	21th March,2023
2.3	Subhajit Majumdar Prakash Singh Debayan Ghosh	All Groups are stored forever. User can opt for Meal while joining a group. UI modified. Basic Validation of Password, Mobile No., Email etc.	10 <sup>th</sup> April,2023

# **1 INTRODUCTION**

This project “NITC TRAVEL TOGETHER” gives users a proper path to travel in a comfortable and affordable way. It is based on making a journey with a group in which the charges of the journey got splitted between the co-passengers. Also we provide so many options such as choosing food according to the user's convenience(veg / non-veg), also a user is allowed to leave from the group before the journey starts and users are allowed to choose a way of entertainment etc. Also there is an option of choosing the group according to the user's need, if the user doesn't want to travel alone.

## **1.1 Document Purpose:**

The purpose of this document is to make an user-friendly interface to give users a proper way to plan their journey in an affordable way within less time of surfing here and there. The document will show how the project is going to work and what are the functionality of the project. It contains some use cases through which users can easily handle all types of test cases.

The motive of this project is to provide a comfortable user experience along with the best pricing available.

## **1.2 Product Scope:**

The scope of the application is to ease the travelling of the user and to create a convenient and easy-to-use application for passengers. The project is based on Java and OOPS concepts and here we also use the database to store the data of our travels. We use the JDBC to make a connection between the java application and the database.

Through this application, many people can know the requirements for their journey and can be fulfilled by our application.





### **1.3 Intended Audience and Document Overview:**

The remaining sections of this document describes the overall descriptions which includes product perspective and functions, characteristics of users. It also consists of Assumptions, and Dependencies. Overall description is listed in section 2. Section 2 mainly comprises Product Overview, Product Functionality, Design and Implementation Constraints, and Assumptions and Dependencies. Section 3 includes Specific Requirements which consists of Functional Requirements, and use case model. Section 4 includes Performance Requirements, Safety and Security Requirements, Software Quality Attributes.

### **1.4 Definitions, Acronyms and Abbreviations:**

There are two basic users for our application: User, and Admin.

- Users can use all the basic features of the app.
- Admin can also access all the features along with an additional feature of uploading the last availability time of the journey, i.e, if the journey is started and then someone wants to join the journey then the admin has the additional feature of removing that journey from the available journeys.

<b>Admin</b>	<b>Administration</b>
<b>IDE</b>	<b>Integrated Development Environment</b>
<b>IEEE</b>	<b>Institute of Electrical and Electronic Engineers</b>
<b>JDBC</b>	<b>Java Database Connectivity</b>
<b>OOP</b>	<b>Object Oriented Programming</b>
<b>SQL</b>	<b>Structured Query Language</b>

## 1.5 Document Conventions: Text convention:

Convention	Description
Heading-Size	18
Sub-heading-Size	17
Text-Size	13
Text-style	Arial
Heading-Style	Bold and underlined with numbered

**Typographic conventions:** Visual Basic documentation uses the following typographic conventions.

<b>SIGN IN, SIGN OUT</b>	Words in capital letters indicate key names and key sequences.
<b>Back, Submit</b>	Objects, the variable name used for goto next page
<b>Empty blocks</b>	indicate placeholders for the information you supply.
<b>First-Name</b>	The first letter capital of such words shows the fields.
<b>View Page</b>	This will indicate the field name which specifies its work.
<b>Dialog box</b>	For any popup, a dialog box shown

## **1.6 References and Acknowledgments:**

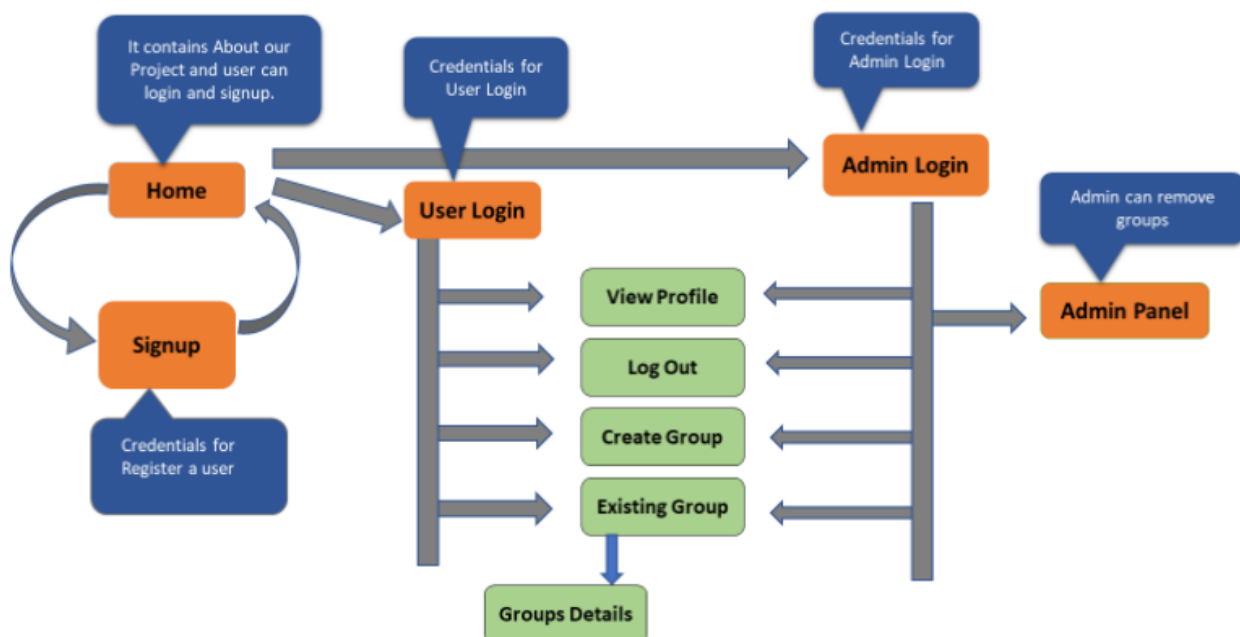
- **Geeks For Geeks**
- **You-tube Videos**
- **Stack Overflow**
- **Object Oriented Programming Books**

## 2 Overall Description

### 2.1 Product Overview:

The overall description will describe the role of the components of the software and their interconnections. Specific requirements will describe the roles and functions of the actors i.e., users, user-admin and admin.

The main objective of the product on “NITC TRAVEL TOGETHER” is to provide facilities to the user to make a journey in a group according to their need. This product can also manage the details of the user from any group and splitting bills. The users are allowed to choose food(veg/non-veg) as per his/her preference, choosing a way of entertainment. The product is built at the administrative end and also a user-admin is guaranteed to access. The purpose of this project is to build an application program to reduce the manual work for managing the user information and all the things related to travelling like fare and making groups and all such things. It tracks all the details of the journey from both the ends ,i.e., the user end and the admin end.



## DATA FLOW DIAGRAM

## **2.2 Product Functionality:**

- **Registration** : Users need to register with his/her name and all other credentials as given in the signup page.
- **Login** : An existing user must login to the system using his/her credentials.
- **Admin Login** : An existing admin can login into the system using his/her credentials.
- **Create Group** : A user can create a group with some credentials and allow others to join his/her group.
- **Add as member** : A user can easily join a group as a member of that group.
- **Opt out from group** : A user can easily leave the group, which he/she joined before.
- **Group details** : Any user can easily traverse the details of any group and join any group as per his need .
- **Remove Group** : Once the group is left for the destination, then admin is allowed to remove that group.
- **Profile** : An existing user can easily see his details after login into the system.
- **Traverse all groups** : An existing user can easily traverse all the groups and can explore them.

## **2.3 Design and Implementation Constraints:**

This product is a desktop based application, so we are using Netbeans here. Java programming language will be used here. For DBMS, we are using mysql. The information of all users, groups and data of all such things are stored in a database that is accessible by the website.

MySQL server will be used as SQL engine and database. The NITC Travel Together is running 24 hours a day. Users may access from any computer that has internet browsing capabilities and an internet connection. Users must have their correct names and password to enter into their online account and do action.

## **2.4 Assumption and Dependencies:**

### **Assumption:**

- The Application is available 24\*7.
- All users are part of NITC and have a NITC mail ID.
- All users are using a device with windows of any version.
- Web connection is reliable.
- While entering the details in the login form, you have to fill all the required details.

### **Dependencies:**

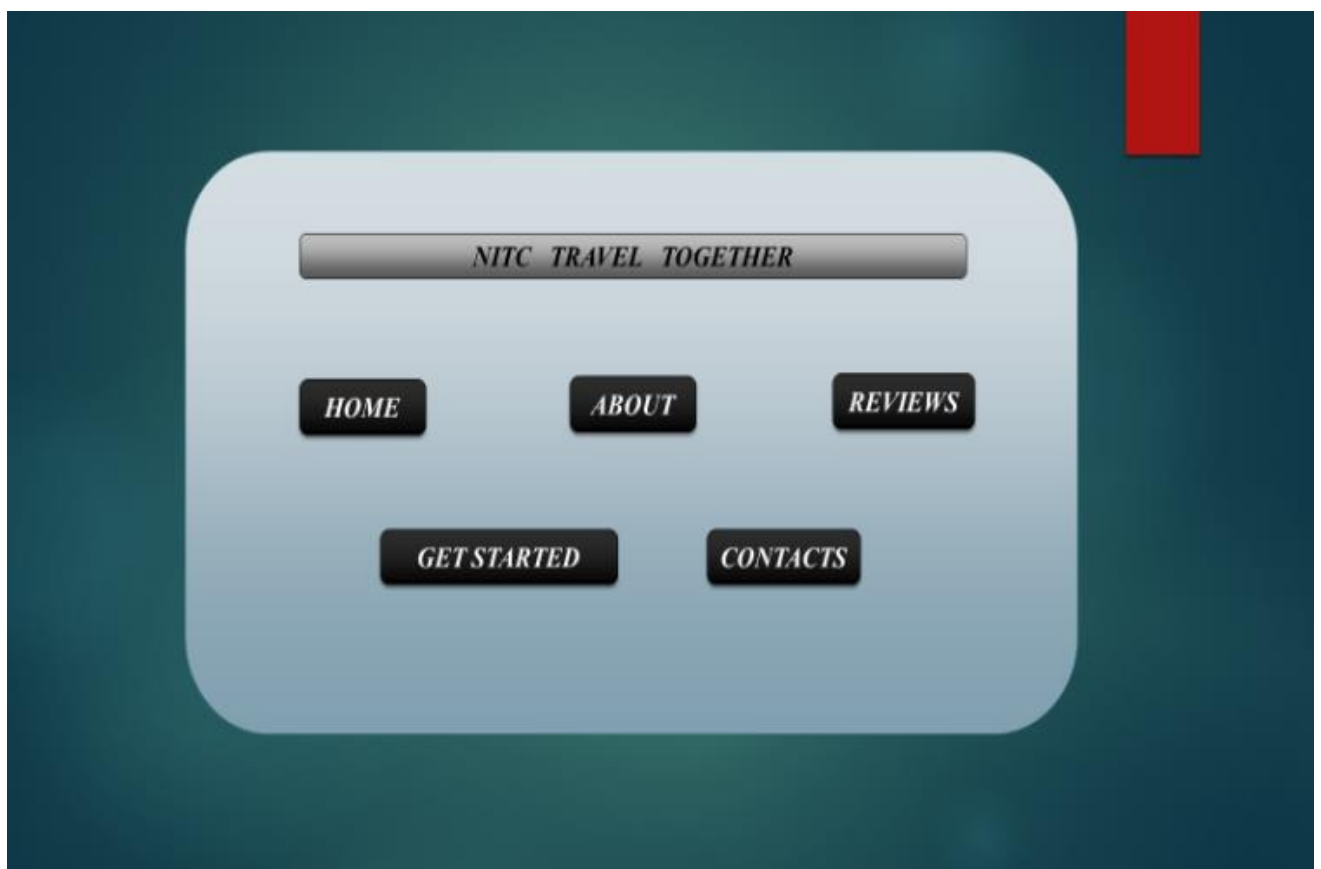
For finding mates to share the travelling fares, there should be mates available who are travelling to the same location or to the locations which lie between the source and destination on the particular day.

### 3. Specific Requirements

#### 3.1 External Interface Requirements

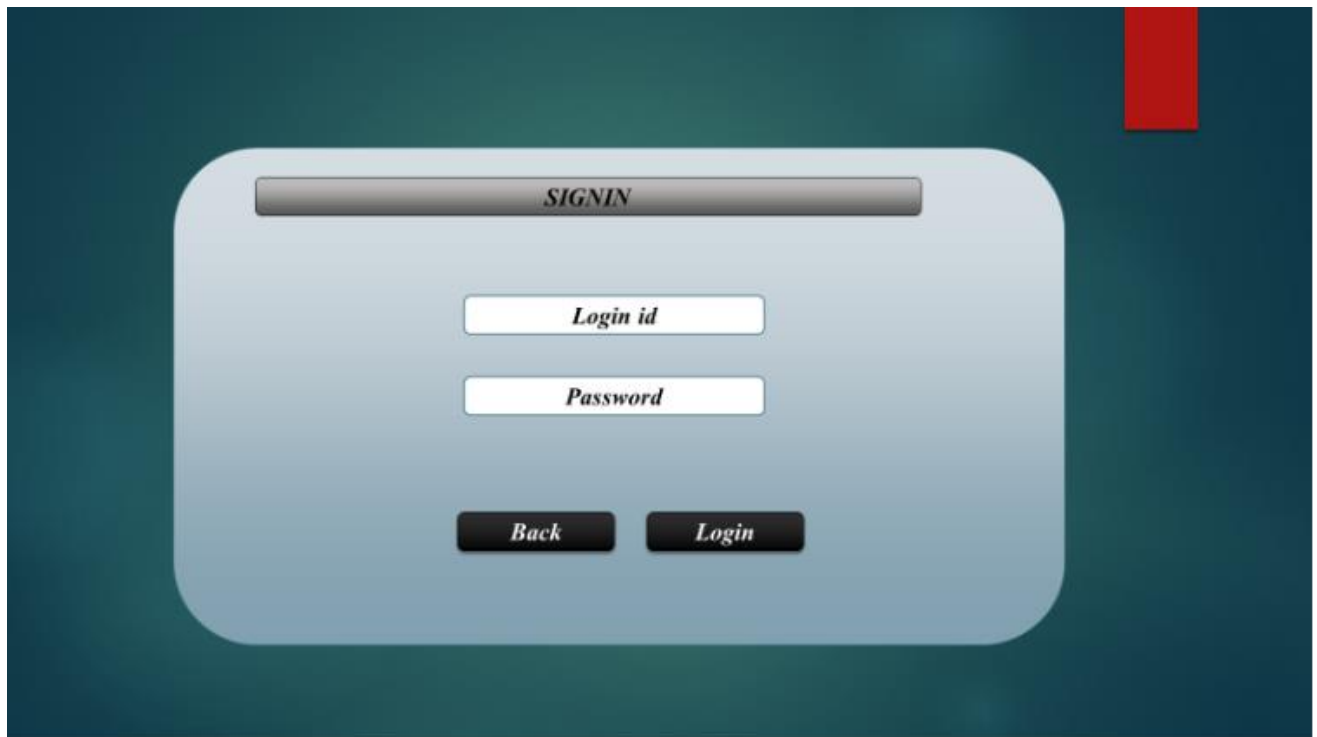
##### 3.1.1 User Interfaces:

**Interface 1:** It is the home page where a user have multiple options to get the information, user can click on 'get started' to login or register in the application, a user can also see the reviews by clicking on 'review' button and a user is allowed to contact us by clicking on 'contact' button, a user can read some data regarding our application by clicking the button 'about'.



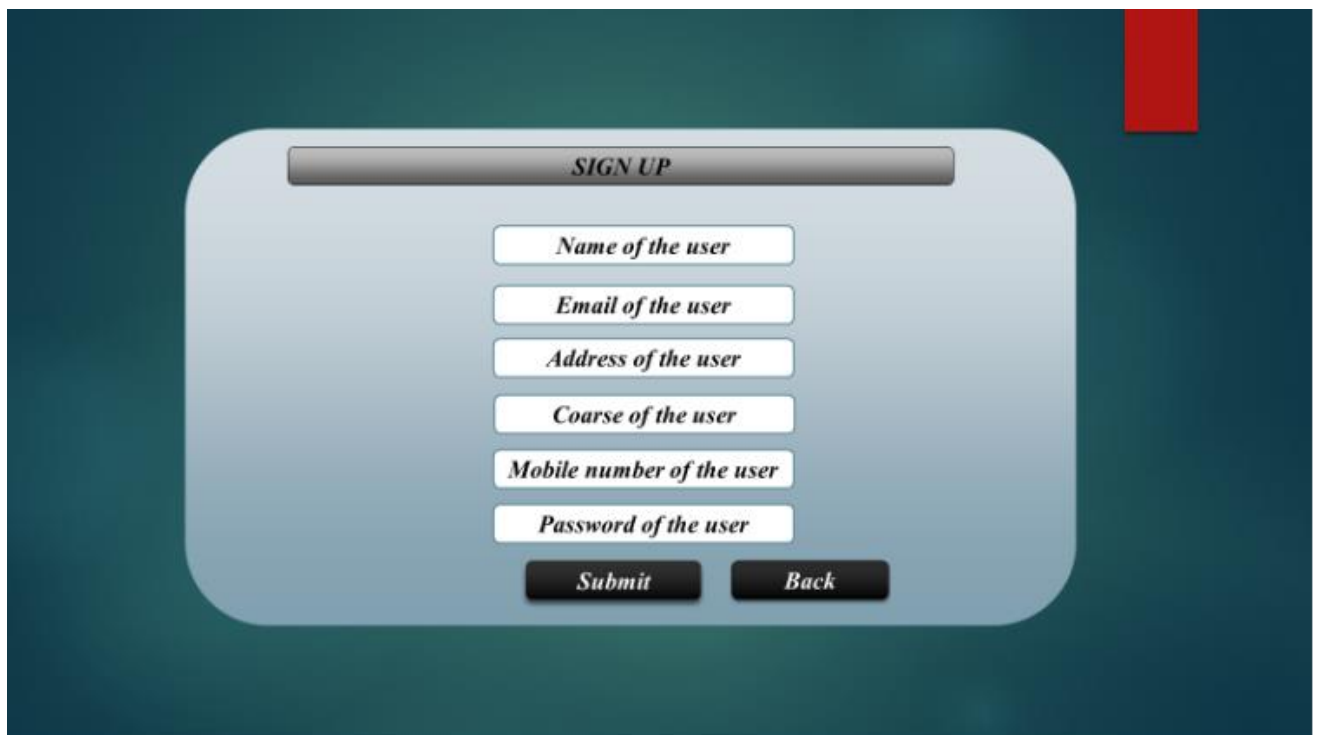
**Interface 2:** This is the sign in page where a user can login in our application. A user is able to sign in if he/she is already registered in our application, otherwise he/she is not allowed to login into the application. If the user is not allowed to login into our application he/she has to register in our application. If he/she already registers then the user has to type the user id which is the email id and the password that the user has created at the time of registration and then the user has to click login to enter into the system.





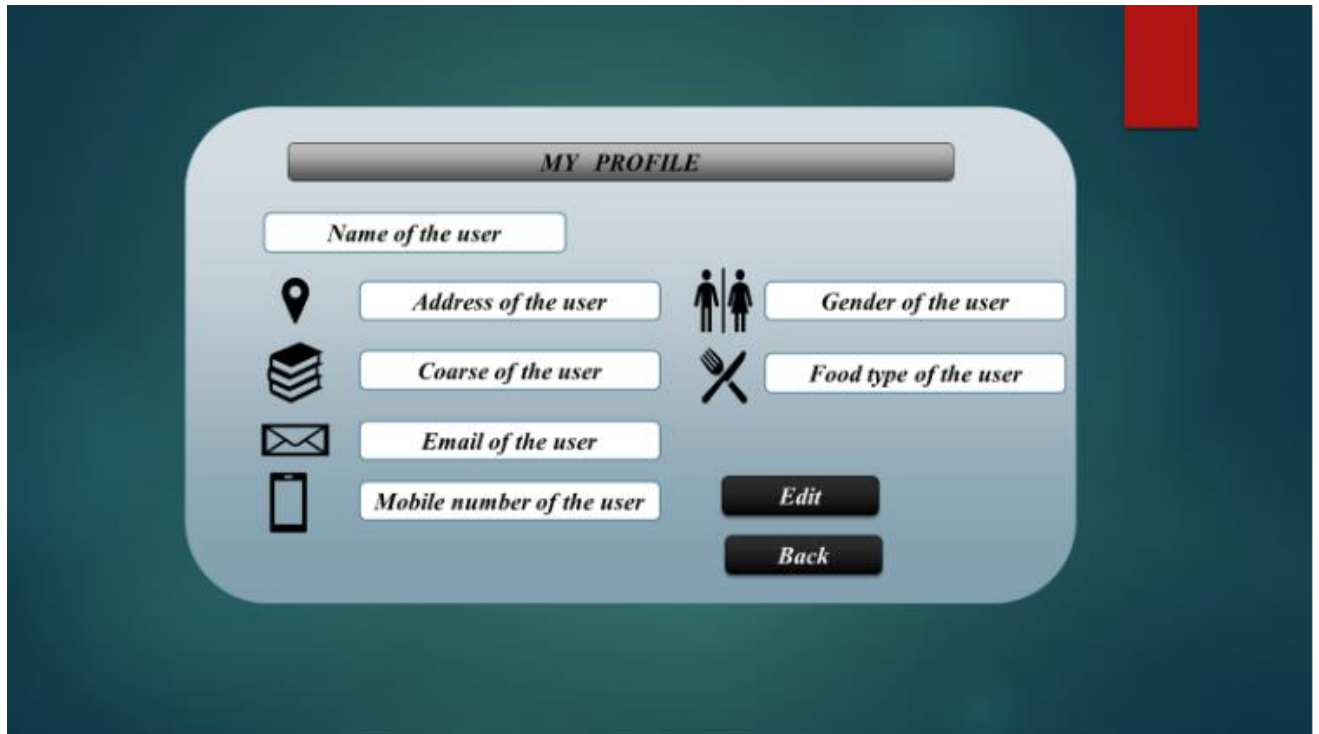
The image shows a 'SIGNIN' interface. It features a light blue rounded rectangle centered on a dark teal background. At the top of the rectangle is a dark grey bar with the word 'SIGNIN' in white. Below this are two white input fields with black text labels: 'Login id' and 'Password'. At the bottom of the rectangle are two dark grey buttons with white text: 'Back' and 'Login'. A red vertical bar is visible on the right side of the dark teal background.

**Interface 3:** This is the signup page which helps users to register in our application. Users have to give some credentials like name ,email id,address,coarse,etc. and then press the submit button to register in our application .



The image shows a 'SIGN UP' interface. It features a light blue rounded rectangle centered on a dark teal background. At the top of the rectangle is a dark grey bar with the words 'SIGN UP' in white. Below this are six white input fields with black text labels: 'Name of the user', 'Email of the user', 'Address of the user', 'Coarse of the user', 'Mobile number of the user', and 'Password of the user'. At the bottom of the rectangle are two dark grey buttons with white text: 'Submit' and 'Back'. A red vertical bar is visible on the right side of the dark teal background.

**Interface 4:** This is the my profile page where the user can see his profile and he can traverse his information like email, education and all such things and can edit his details by clicking the edit button.



The screenshot displays a 'MY PROFILE' interface on a dark teal background. A light blue rounded rectangle contains the profile details. At the top is a header bar labeled 'MY PROFILE'. Below it, the 'Name of the user' is shown in a white box. To the left of the 'Address of the user' box is a location pin icon. To the right of the 'Address of the user' box is a gender icon (male and female figures) and the 'Gender of the user' box. Below the 'Address of the user' box is a 'Coarse of the user' box with a stack of books icon to its left. To the right of the 'Coarse of the user' box is a 'Food type of the user' box with a fork and knife icon to its left. Below the 'Coarse of the user' box is an 'Email of the user' box with an envelope icon to its left. Below the 'Email of the user' box is a 'Mobile number of the user' box with a mobile phone icon to its left. At the bottom right of the profile box are two dark buttons labeled 'Edit' and 'Back'.

**Interface 5:** This is the create group page where a user can become the admin of his group by creating his group by giving some information regarding his group like size, total fare and destination etc.



The screenshot displays a 'CREATE GROUP' interface on a dark teal background. A light blue rounded rectangle contains the group creation fields. At the top is a header bar labeled 'CREATE GROUP'. Below it, there are five white input boxes stacked vertically, labeled 'From where journey ends', 'From where journey starts', 'Total number of persons', 'Means of transport', and 'Total fare of the group'. At the bottom of the form are two dark buttons labeled 'Back' and 'Create'.

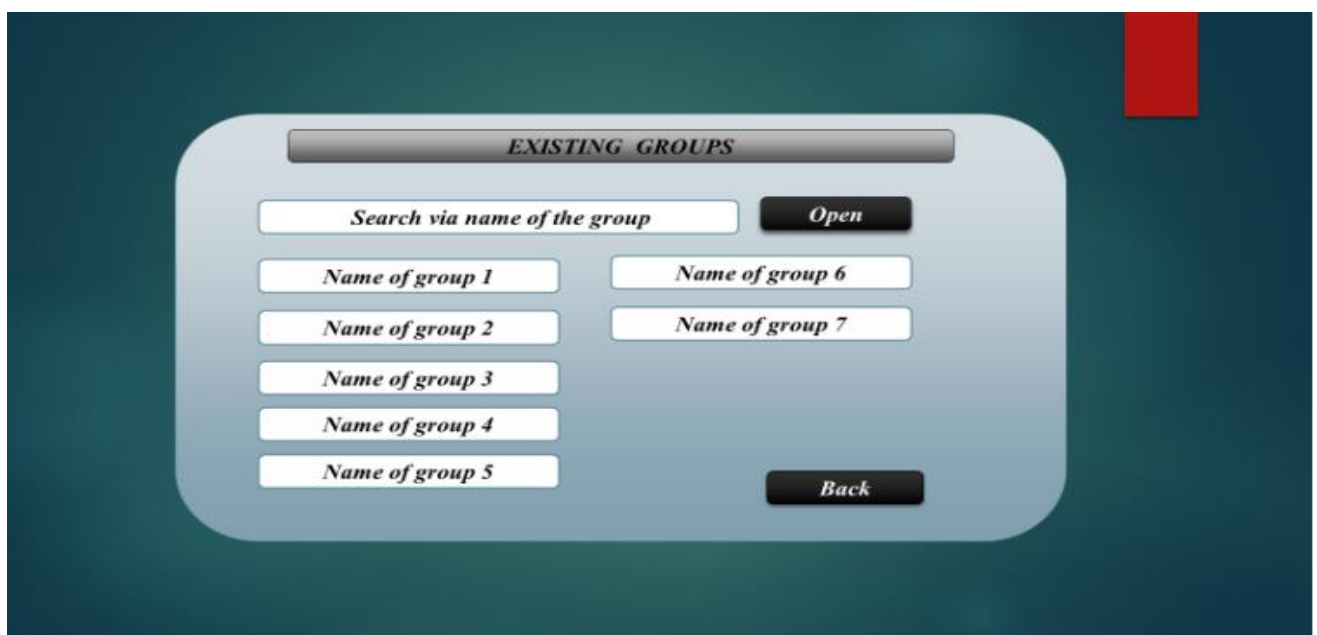
**Interface 6:** This is the group details page where the user can see the details of the group he created earlier, like total member and destination and total fare.

The user can add in the group and also opt out from the group .



The screenshot shows a mobile application interface titled "GROUP DETAILS". It features a light blue rounded rectangle on a dark teal background. Inside the rectangle, there are several input fields and buttons. The input fields are labeled: "Name of the admin", "Size of the group", "From where journey starts", "From where journey ends", "Means of transport", and "Total fare of the group". To the right of these fields are buttons labeled "Split bills", "pay", "Travel alert", "Meal", and "Entertainment". At the bottom of the rectangle are three buttons: "Back", "Add", and "Optout". A red vertical bar is visible on the right side of the screen.

**Interface 7:** This is the existing groups page where the user can see all the groups and the user can open any group via the group name and then the user can add into the group and also opt out from the group.



The screenshot shows a mobile application interface titled "EXISTING GROUPS". It features a light blue rounded rectangle on a dark teal background. Inside the rectangle, there is a search bar labeled "Search via name of the group" and an "Open" button. Below the search bar, there are five input fields labeled "Name of group 1", "Name of group 2", "Name of group 3", "Name of group 4", and "Name of group 5". To the right of these fields, there are two input fields labeled "Name of group 6" and "Name of group 7". At the bottom right of the rectangle is a "Back" button. A red vertical bar is visible on the right side of the screen.

**Interface 8:** This is the admin panel page where admin can remove the group from the existing group, as the group leaves the starting location.



### 3.1.2 Hardware Interfaces:

No additional hardware interface is needed. A computer device capable of running a desktop application is sufficient.

### 3.1.3 Software Interfaces:

Software presented in this SRS does not need any other software interface than the operating system itself.

## 3.2 Functional Requirements:

**F1 :** The system shall allow the user to access all the fields.

- **Description :** It is a home page which has Home, About, Reviews, Get Started, and Contact button.
- **Input :** User can click on any button for further information.
- **Processing :** System will allow the user to enter on that particular page with credentials, if required.

- **Output :** User is prompted to the page, he/she clicked.

**F2 :** The system shall allow the user to sign in.

- **Description :** Sign In page for the users .
- **Input :** Users enter their login IDs and password and click on the login button.
- **Processing :** System will authenticate the user with its credentials.
- **Output :** After successful authentication, the system prompts the user to next step what he/she wants .

**F3 :** The system shall allow the user to register themselves.

- **Description :** Register page for users.
- **Input :** Users have to enter the following credentials:- Name, Course, Address, Email, Phone number, Age, Gender, Password, etc.
- **Processing :** Application sends the entered information to the server and server verifies it.
- **Output :** After a successful registration, the user can go to the next step.

**F4 :** The system shall allow the user to edit one's Profile.

- **Description :** If anything the user wants to update, can update through this.
- **Input :** After clicking on the edit button, the system will allow the user to edit the profile.
- **Processing :** Users profile is updated as per his need.
- **Output :** Updates profile with all the changes.

**F5 :** The system shall allow the user to create a group.

- **Description :** Users can create a group for the journey.
- **Input :** Users have to give credentials like Travel Location, Vehicle, Total Member, Total fare, etc.
- **Processing :** System will create a group with those information users gave.
- **Output :** A group is created successfully.

**F6 :** The system shall allow the user to see all the groups which exist.

- **Description :** For displaying the groups that were created already.

- **Input :** Users have to click on the open button which redirects users to the group details .
- **Processing :** System will display the groups which exist.
- **Output :** Displays the existing groups successfully.

**F7 :** The system shall allow the user to display the group details of a particular group.

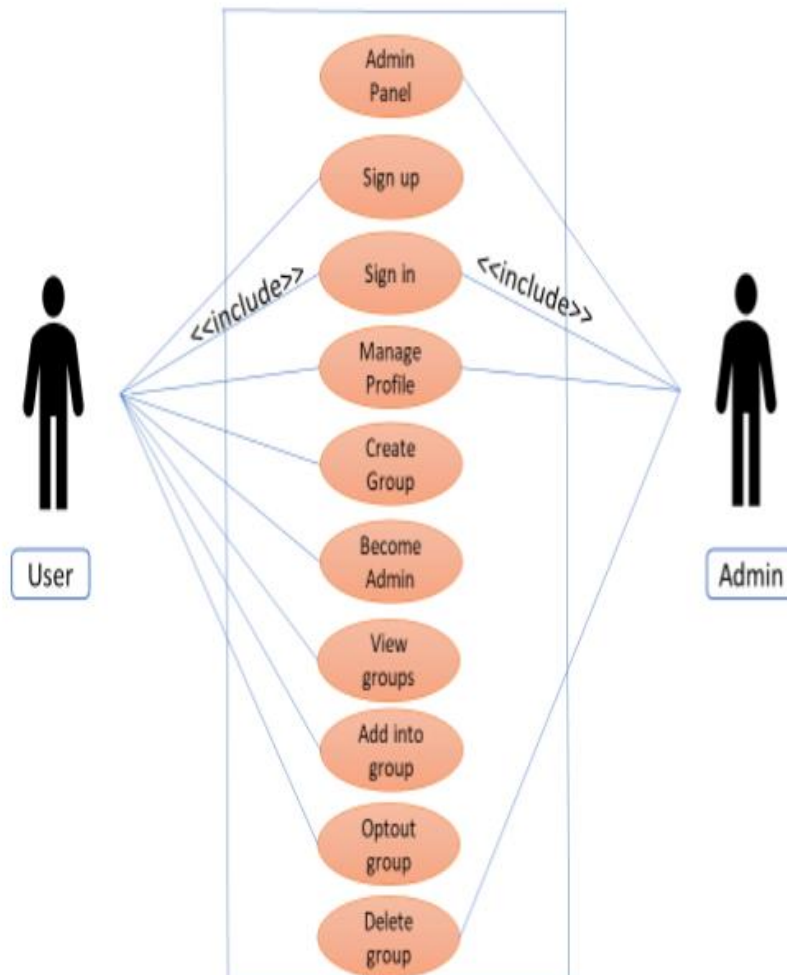
- **Description :** User can display the group details of a particular group.
- **Input :** Users have to give the name of the admin of that particular group.
- **Processing :** System will display the details, if any group exists.
- **Output :** Details of that group displayed.

**F8 :** The system shall allow the admin to remove groups from the existing groups .

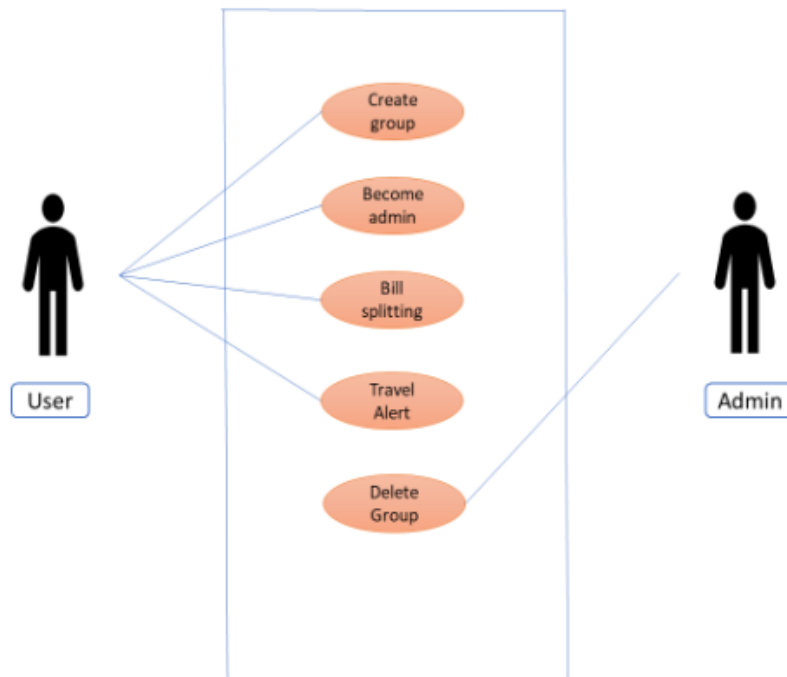
- **Description :** Admin can remove the groups when the journey is completed.
- **Input :** Group name with leaving date is required for removing the group.
- **Processing :** The system will allow the admin to remove the group.
- **Output :** Group is removed successfully.

### 3.3 Use case model :

#### Use case model 1 :



## Use case model 2 :



### 3.3.1 Use case 1:

**Author :** Riya

**Purpose :** To allow the user to register in an application by giving some information .

**Requirement Traceability :** F3

**Priority :** High

**Pre Condition :** The user doesn't have an account prior to application .

**Post Condition :** None

**Actor :** User

**Flow of Event :** User fills all information and clicks on signup button , then all data given by user store in the database .



### **3.3.2 Use case 2:**

**Author :** Khushboo.

**Purpose :** To allow the user to login in an application by some credentials.

**Requirement Traceability :** F2

**Priority :** High

**Pre Condition :** The user must be a registered user .

**Post Condition :** None

**Actor :** User

**Flow of Event :** User can sign in using user id and password and then click on signin button .

### **3.3.3 Use case 3:**

**Author :** Amay .

**Purpose :** To create a group by entering some information .

**Requirement Traceability :** F5

**Priority :** High

**Pre Condition :** User must sign in .

**Post Condition :** group must be present in the existing group page.

**Actor :** User

**Flow of Event :** User can click on create group and fill all information and then click on create button .

### **3.3.4 Use case 4 :**

**Author :** Khushboo

**Purpose :** To allow the admin to login into the system .

**Requirement Traceability :** F8

**Priority :** High

**Pre Condition :** Admin knows the id and the password.

**Post Condition :** None

**Actor :** Admin

**Flow of Event :** Admin can give the id and password and click on the login button .

### **3.3.5 Use case 5 :**

**Author :** Riya

**Purpose :** User can see all the existing groups and can search for a particular group to see that group's details .

**Requirement Traceability :** F6

**Priority :** Medium

**Pre Condition :** A user must login in the application and the group he is looking for , must be in the existing group .

**Post Condition :** None

**Actor :** User

**Flow of Event :** User clicks on existing group button and then searches for the group the user is looking for and then clicks open .

### **3.3.6 Use case 6 :**

**Author :** Amay

**Purpose :** To add users in a group or opt out from the group or add meal or entertainment during journey and splitting bills .

**Requirement Traceability :** F7

**Priority :** High

**Pre Condition :** A user must login in the application and open the group details page of that particular group.

**Post Condition :** User must be added in the group and also the other activity be done .

**Actor :** User

**Flow of Event :** Users must click on add to join the group , opt out for removal from the group , meal for adding meal , entertainment for adding entertainment and split bills for splitting bills .

## **4. Other Non-Functional Requirements**

Functional requirements define the needs in terms of performance logical database Requirements, design constraints, reliability, availability, security, maintainability.

### **4.1 Performance Requirements :**

Performance requirements define acceptable response times for system functionality.

1. The load time for the user interface screen shall take no longer time .
2. The login information shall be verified within a few seconds.
3. Queries shall return the result in a few seconds.
4. The performance of the system will highly depend on the performance of the hardware and software components of the installed computer.

### **4.2 Safety and Security Requirements :**

There are several user levels in the NITC Travel Together. This gives different views and accessible functions of user levels through the system. Maintaining backups ensures the system database security.

Admin will be able to login to the NITC Travel Together. Customer Service Representatives(User Admin) will have access to make a journey. Admin will have access to the Management subsystem. Also Admin will have the access to cancel the journey as if needed.

### **4.3 Software Quality Attributes :**

- **Availability:** - The system shall be available 24 hours.
- **Correctness:** - Extent to which program satisfies specifications, fulfils user's mission Objectives.
- **Efficiency:** - Users can make a journey anytime , anywhere.
- **Validation:** - System will verify the credentials of the Admin.
- **Portability:** - NITC Travel Together shall run in any web browser.

## Appendix A - Activity Log

<b>18 Jan 2023</b>	<b>Discuss on topic selection and how to divide the work among us</b>
<b>20 Jan 2023</b>	<b>Meet regarding requirement of the project and front end design</b>
<b>21 Jan 2023</b>	<b>Evaluate the progress and database design in meet</b>
<b>22 Jan 2023</b>	<b>SRS Documentation</b>
<b>23 Jan 2023</b>	<b>Completing the SRS Documentation and defining the Use Case diagram.</b>

<Leave this page blank>

---

# Design Document

for

## <NITC TRAVEL TOGETHER>

Version<2.3>

Prepared by Team 10:  
(Based on SRS Version 1.1 prepared by Team 7)

KARAN SACHINKUMAR  
SATISH

M210687CA

karan\_m210687ca@nitc.ac.in

ATUL KUMAR SETHI

M210663CA

atul\_m210663ca@nitc.ac.in

SHIVAM KUMAR JHA

M210702CA

shivam\_m210702ca@nitc.ac.  
in

**Project Owner:** ADHIKARI VENKATA YUKTHA

**Course:** CS4096 Software Engineering Laboratory

**Date:** 07-02-2023





## Glossary

<b>Admin</b>	<b>Administration</b>
<b>SQL</b>	<b>Structured Query Language</b>
<b>ER</b>	<b>Entity Relation</b>
<b>ID</b>	<b>Identity</b>
<b>UML</b>	<b>Unified Modeling Language</b>
<b>JFC</b>	<b>Java Foundation Classes</b>
<b>API</b>	<b>Application Programming Interface</b>
<b>GUI</b>	<b>Graphical User Interface</b>
<b>AWT</b>	<b>Abstract Window Toolkit</b>
<b>RDBMS</b>	<b>Relational Database Management System</b>
<b>TC</b>	<b>Test Case</b>

## Table of contents

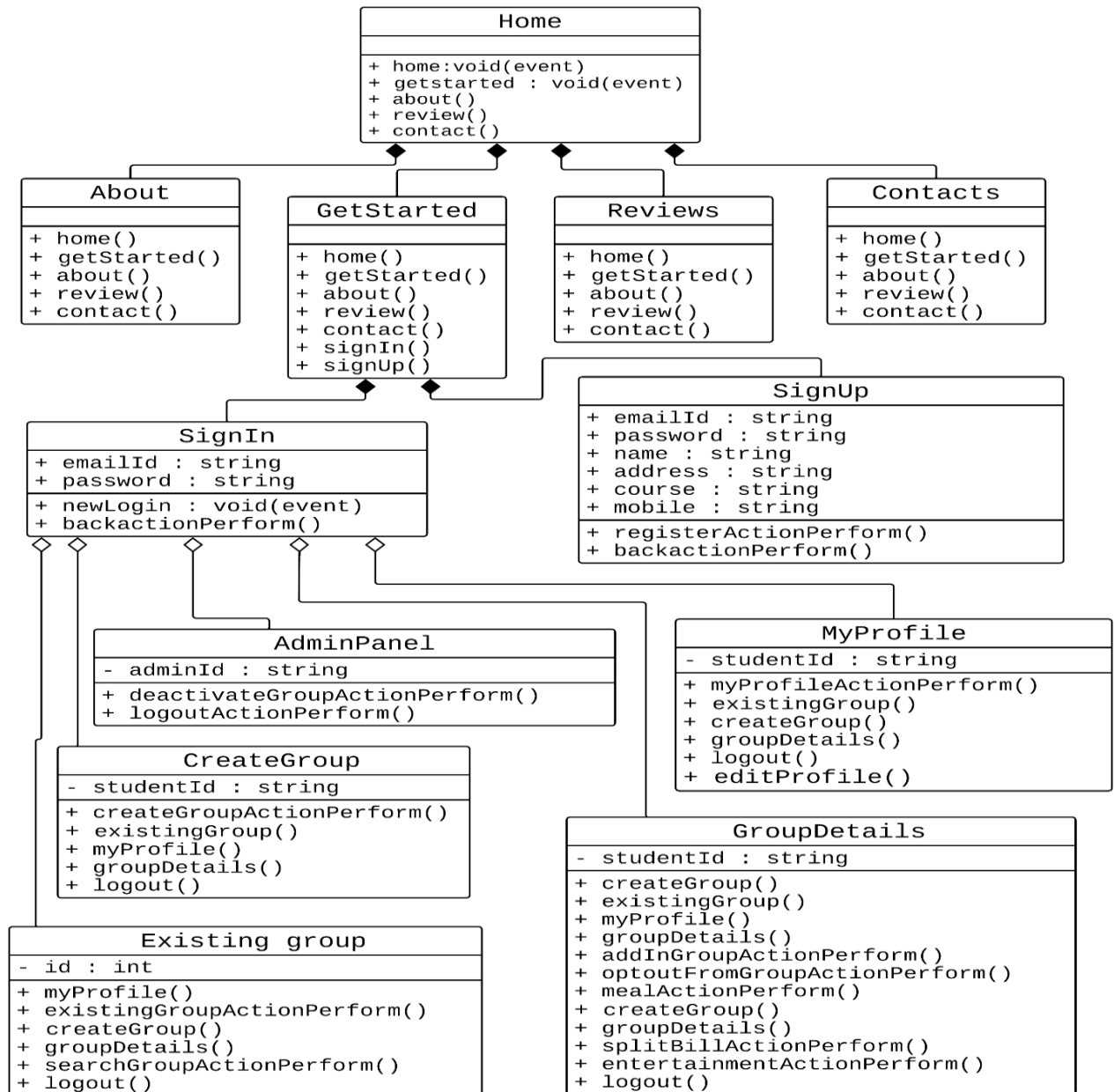
<b>Appendix A - Activity Log</b>	29
Glossary	33
<b>Table of contents</b>	34
1. Detailed Design through UML diagrams	35
1.1 System model using Class Diagram	35
1.1.1 Class Diagram	35
1.2 Responsibilities - Use Case Diagram	37
1.3 Static snapshot of the system - Object Diagram	38
1.4 System Interactions through Sequence Diagrams	39
1.4.1 Sequence Diagram 1 for SignUp /Sign-in	39
1.4.2 Sequence Diagram 2 for Creating Group	40
1.4.3 Sequence Diagram 3 for joining the existing group	41
1.4.4 Sequence Diagram 4 for Deactivating the Group by Admin	42
1.5 Control and Data Flows through Activity Diagrams	43
1.5.1 Activity Diagram 1 for SignIn OR SignUp	43
1.5.2 Activity Diagram 2 for participating in the group	44
1.5.3 Activity Diagram 3 for Admin Deactivating Group	45
2. Database Design	46
2.1 ER Diagram	46
3. Implementation Plans	47
3.1 Technology Stack	47
3.2 User Interface Prototyping	47
<b>4.1 Test Case #1 (User Registration TC1)</b>	53
<b>4.2 Test Case #2 (User Login TC2)</b>	54
<b>4.3 Test Case #3 (User Registration TC3)</b>	54
<b>4.4 Test Case #4 (Create Group TC4)</b>	55
<b>4.5 Test Case #5 (Create Group TC5)</b>	55
<b>4.6 Test Case #6 (Optout from group TC6)</b>	56
<b>4.7 Test Case #7 (Deleting the group TC7)</b>	57
5. Traceability Matrix	58
References	59

# 1. Detailed Design through UML diagrams

## 1.1 System model using Class Diagram

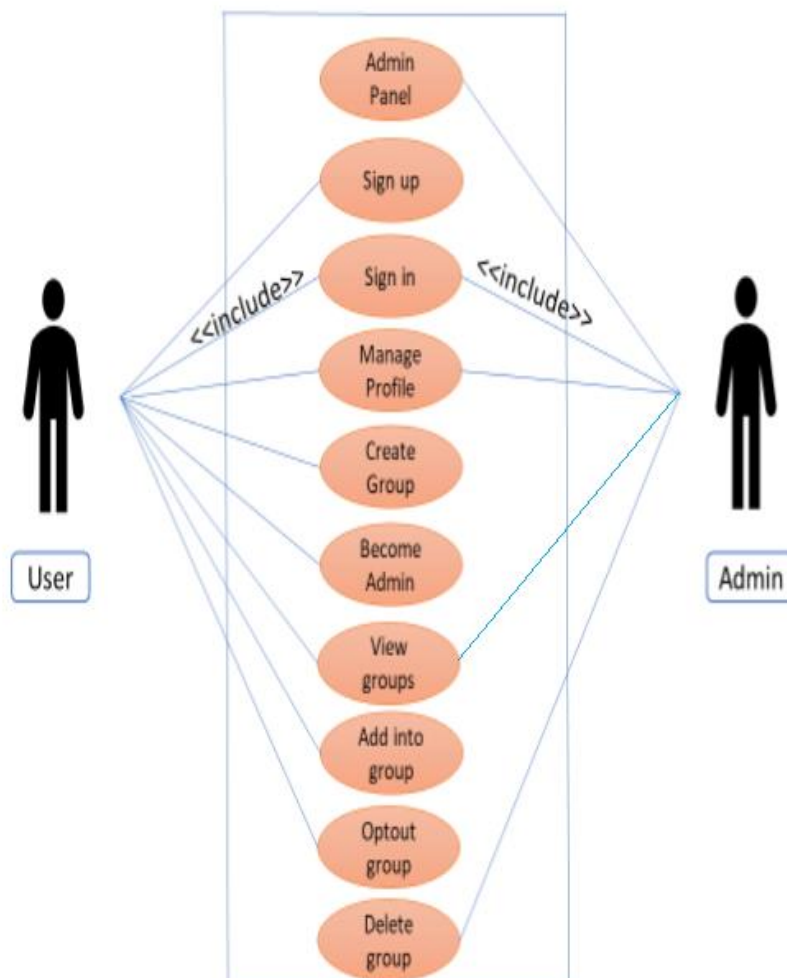
Class Diagram in the Unified Modelling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods) and the relationships among classes.

### 1.1.1 Class Diagram



## Class Diagram

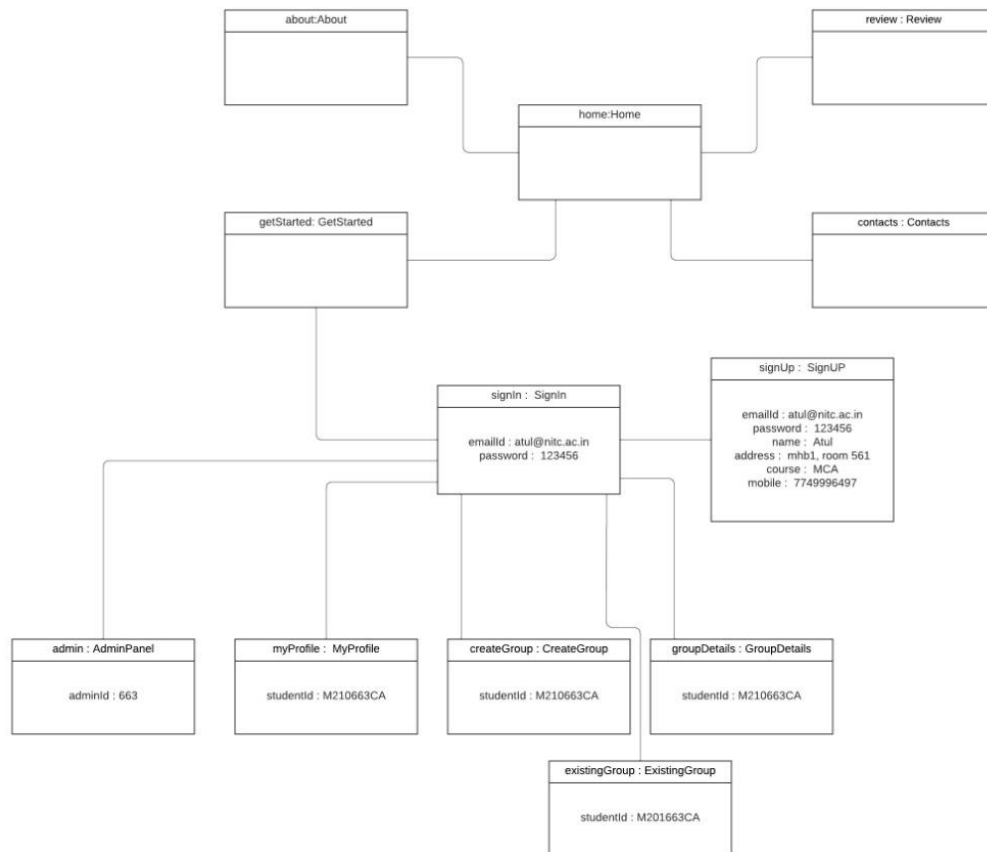
## 1.2 Responsibilities - Use Case Diagram



---

Updated Use Case Diagram

## 1.3 Static snapshot of the system - Object Diagram

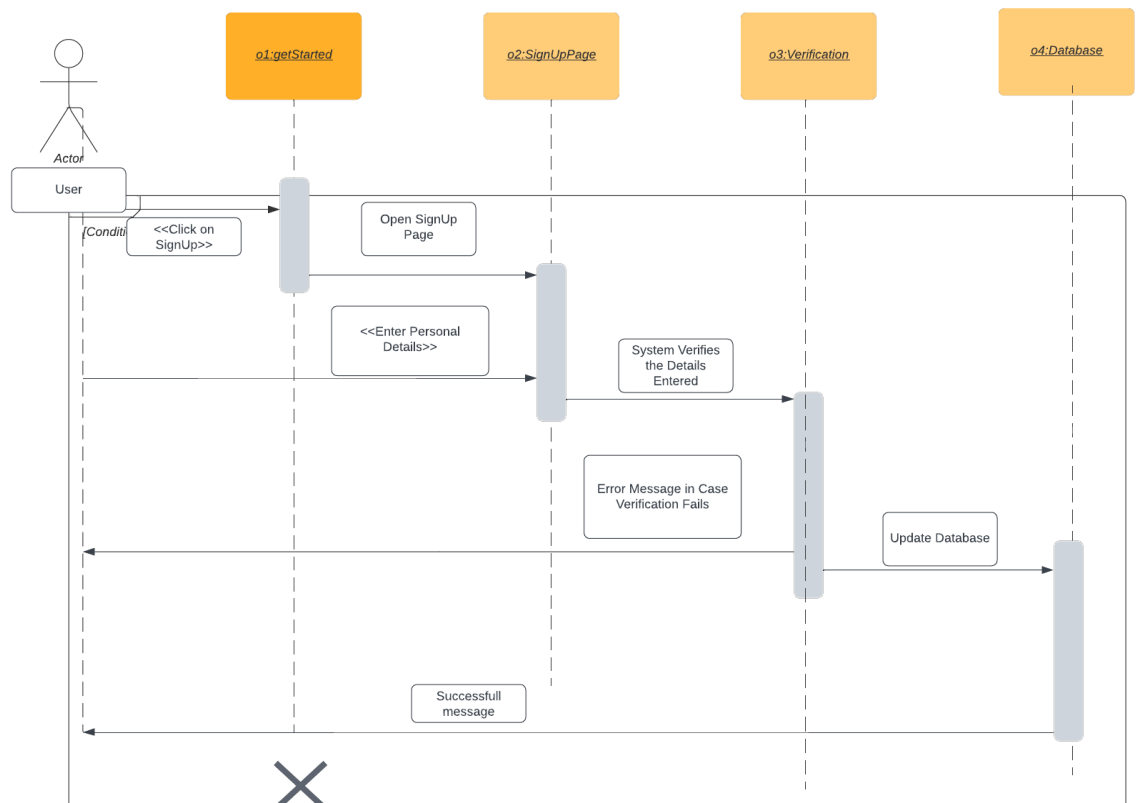


## 1.4 System Interactions through Sequence Diagrams

Sequence diagrams are interaction diagrams that show the sequence of messages exchanged by the set of objects performing a certain task. A sequence diagram shows, as parallel vertical lines (lifeline), different processes or objects that live simultaneously, and as horizontal arrows, the messages exchanged between them, in the order in which they occur.

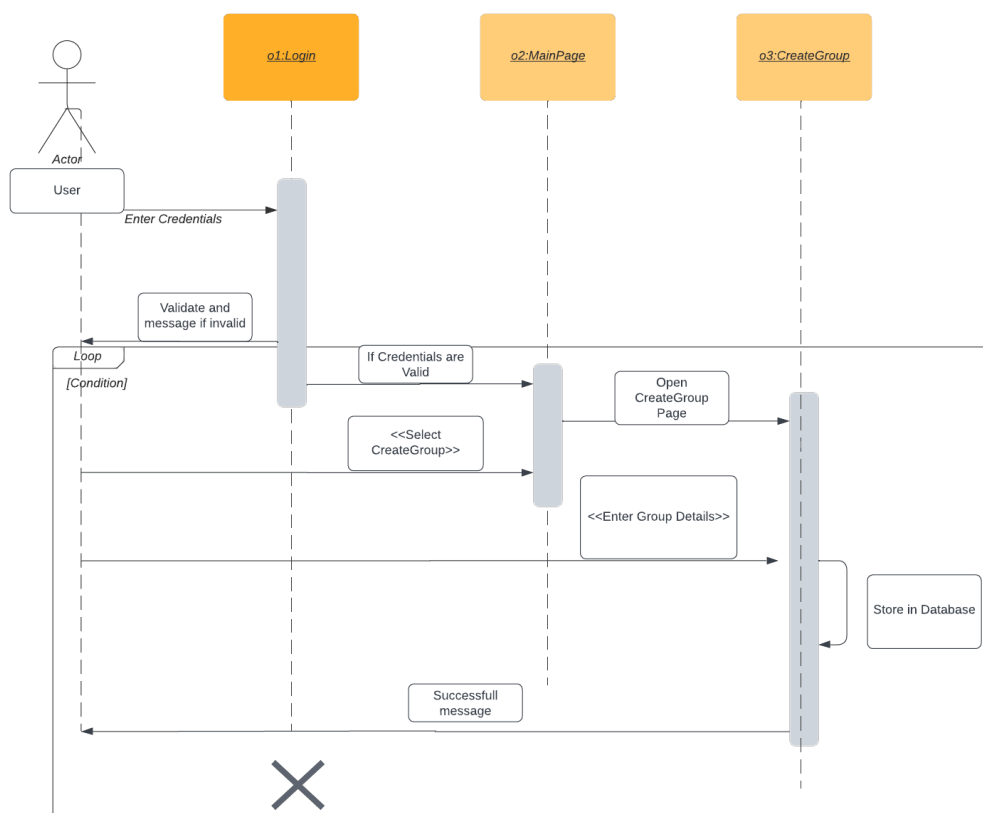
### 1.4.1 Sequence Diagram 1 for SignUp /Sign-in

This Sequence diagram shows how the verification and database update will be done during SignUp. User on the getStarted page clicks on the sign-up option to open the Signup Page. There the user enters the required details and verification of data entered is done and if valid data is entered then User successfully gets registered and necessary changes are done in the database.



### 1.4.2 Sequence Diagram 2 for Creating Group

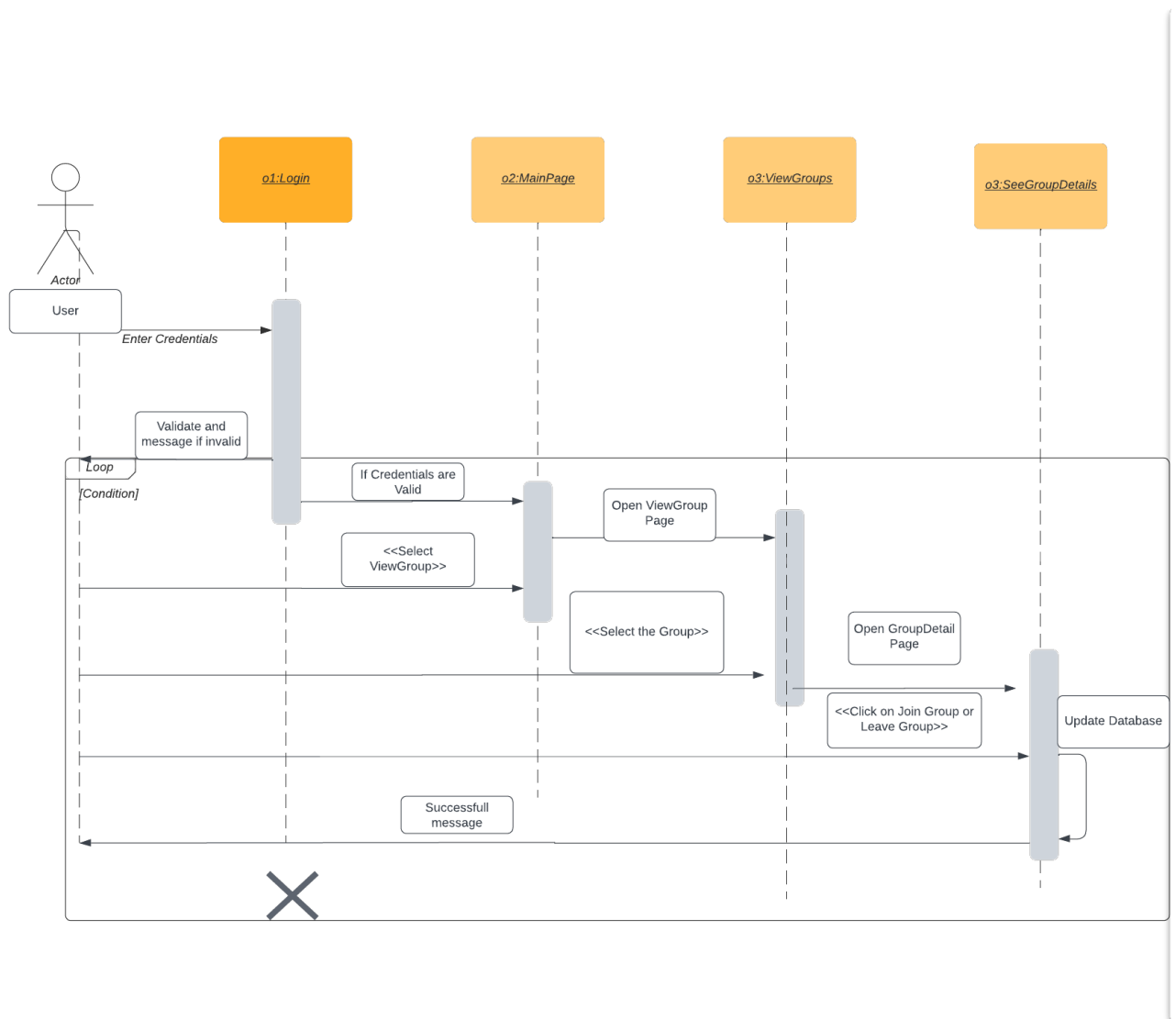
This Sequence diagram shows how a new group is created to be visible for other members. Users must first be logged in to create a group. After logging in, the user is directed to the Mainpage and from there the user clicks on Create group. The user is now directed to the CreateGroup Page, then on entering the group details the user can create a group and gets a successful message.





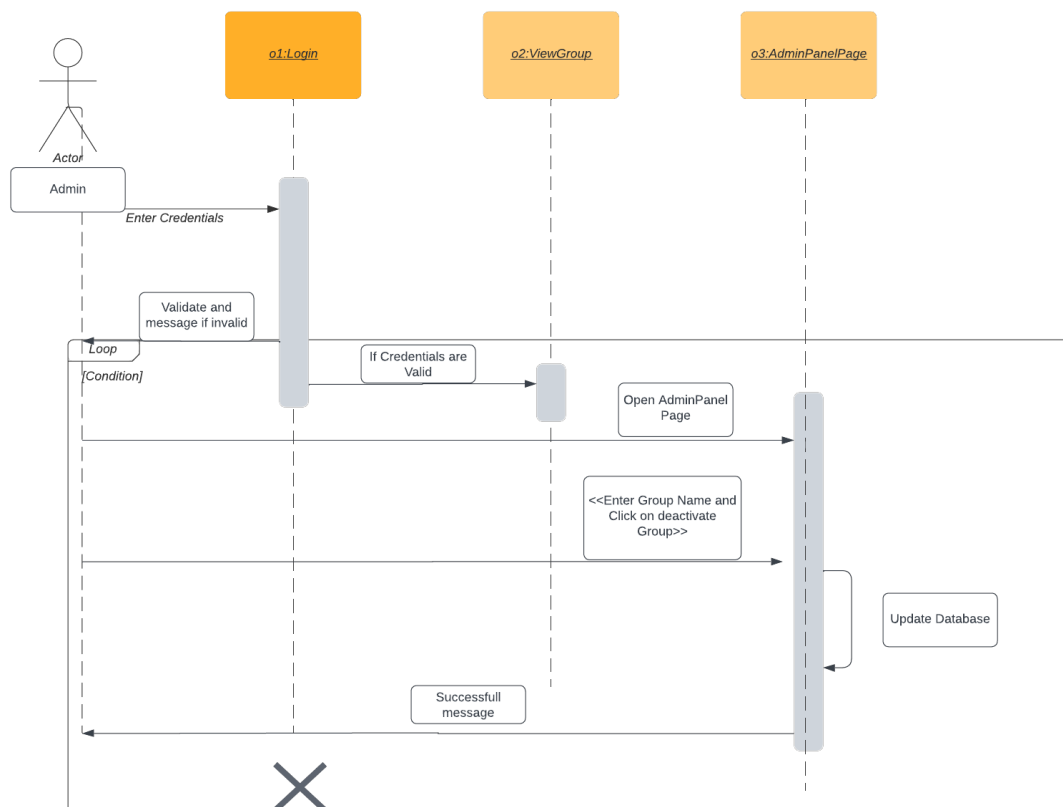
### 1.4.3 Sequence Diagram 3 for joining the existing group

This Sequence diagram shows how users can join the already existing group. Users must first be logged in to join a group. After Logging in, the user is directed to Mainpage and from there User can navigate to ViewGroup page there user can view all groups and can even search for a group then selecting the group user can click on join to join the group.



#### 1.4.4 Sequence Diagram 4 for Deactivating the Group by Admin

This Sequence diagram shows how the Admin can deactivate the group after the trip has reached the destination. Admin must first be logged in to deactivate the group. After logging in admin can view all the groups that already reached the destination. Then on AdminPanelPage admin can enter Group name and click on deactivate.

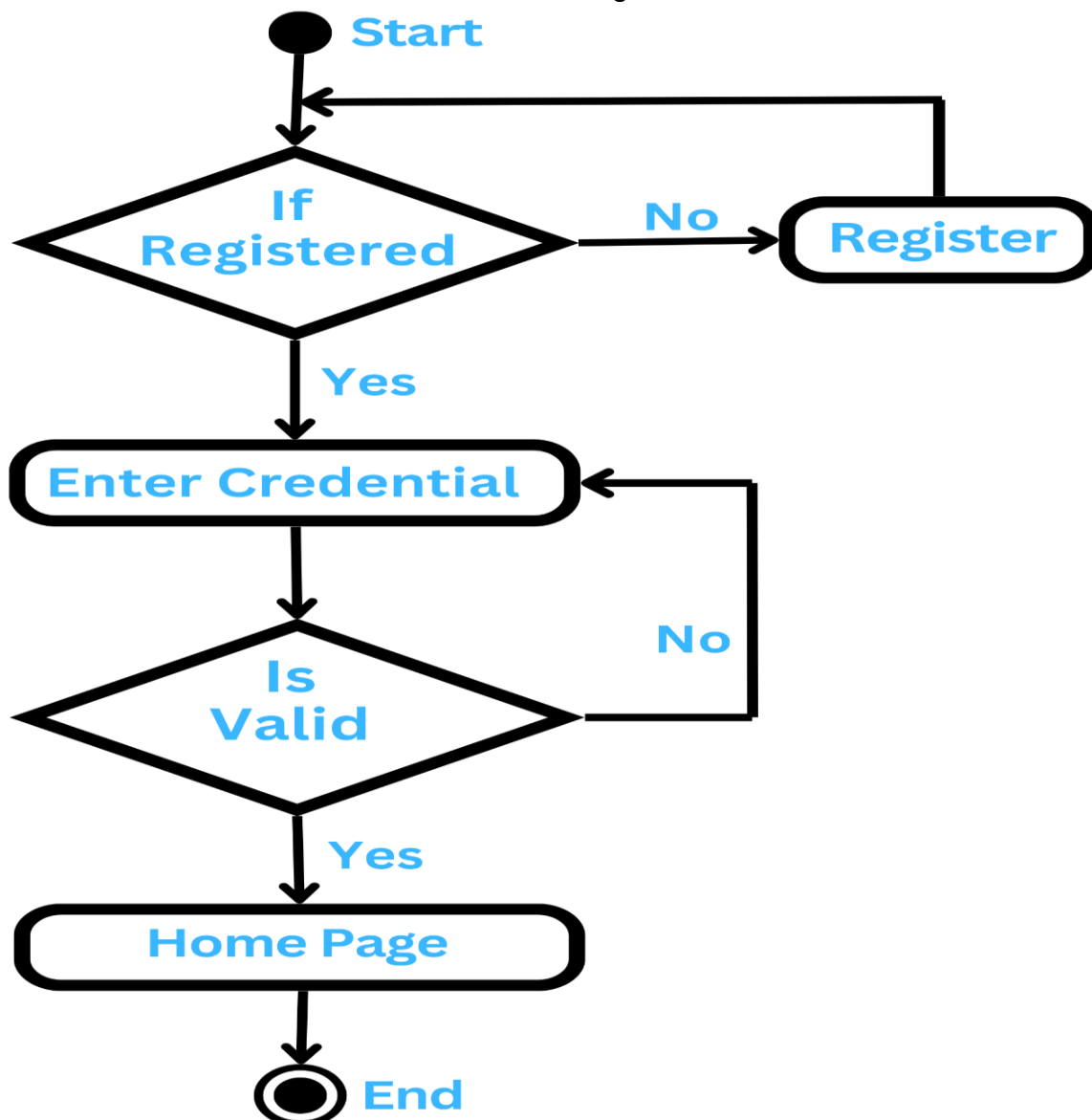


## 1.5 Control and Data Flows through Activity Diagrams

Activity diagram is another important behavioural diagram in UML diagram to describe dynamic aspects of the System. Activity diagram is essentially an advanced version of Flow Chart that modelling the flow from one activity to another.

### 1.5.1 Activity Diagram 1 for SignIn OR SignUp

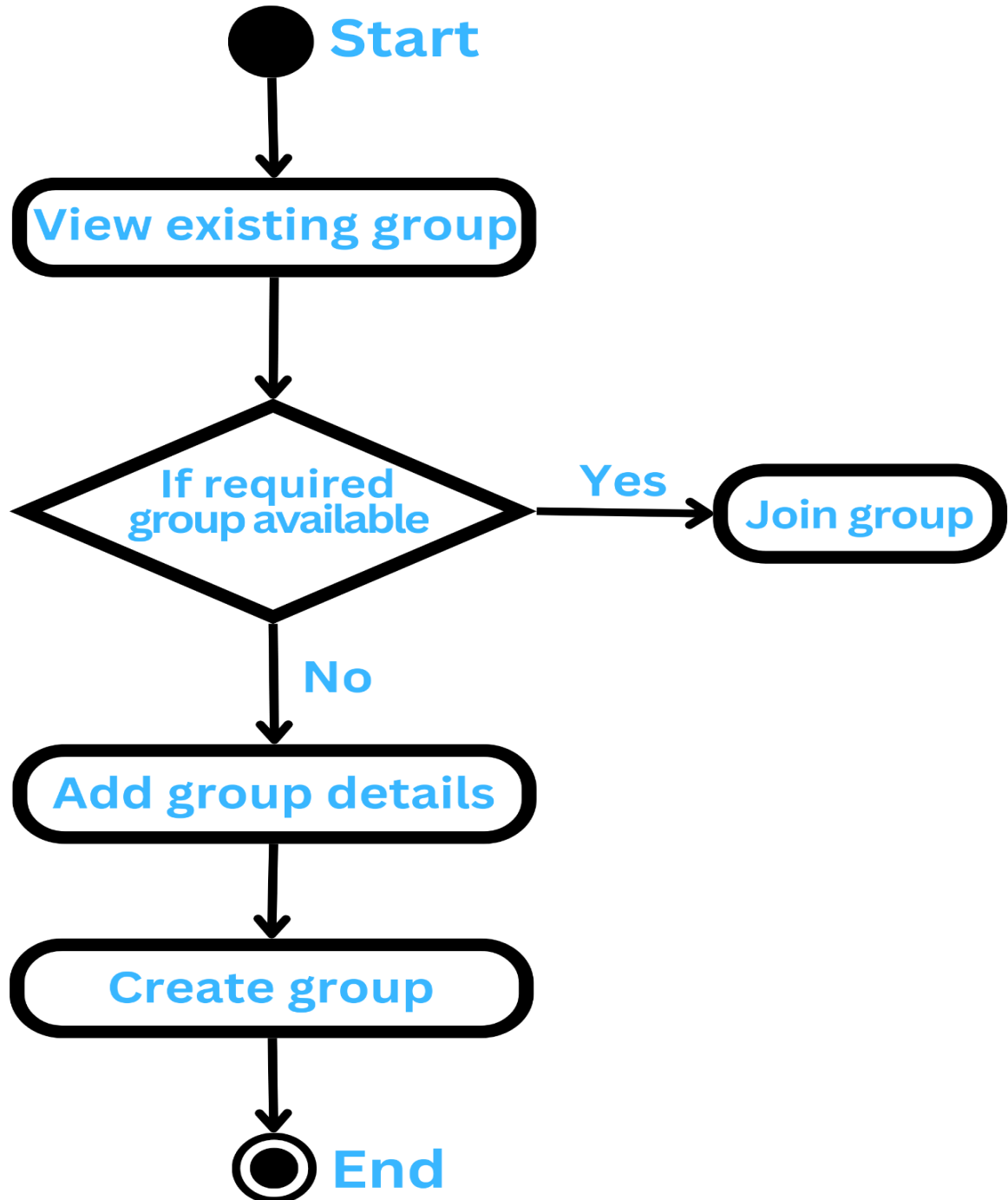
Activity Diagram showing the flow of activities performed for signup or if already registered then sign in. Users must be registered to sign in. If a user is registered then the user has to enter the credentials after which the authentication is done. After successful authentication, the user is taken to the Home Page.



**Activity diagram for Sign up & Sign in**

### 1.5.2 Activity Diagram 2 for participating in the group

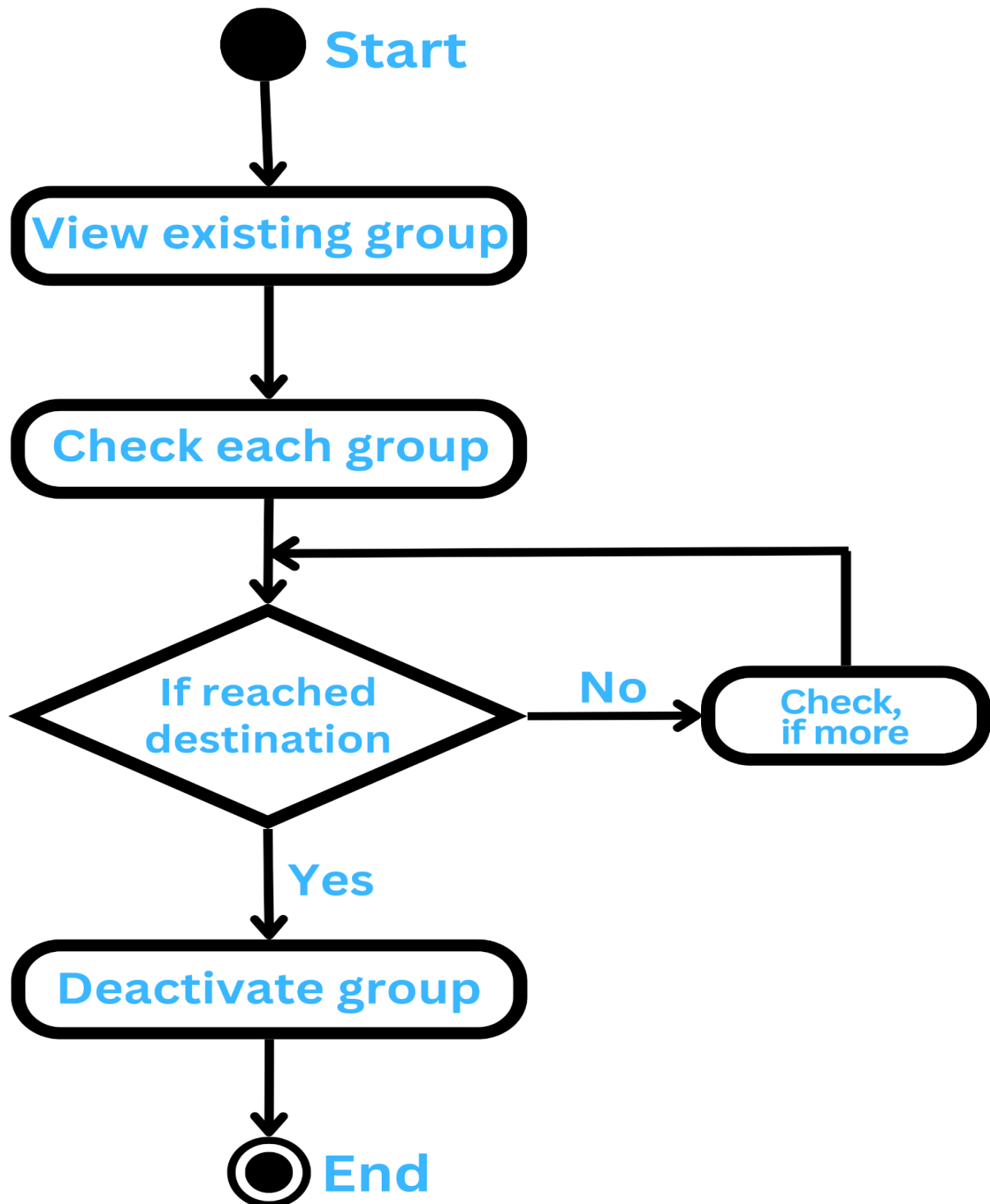
Activity Diagram showing the flow of activities performed for participating in a group by joining an existing group or by creating a new group. There is Two ways to participate in a group. If there is already a group with required source and destination is already created then users can join that group or else users can create a new group by entering the required details.



**Activity diagram for User participating in group**

### 1.5.3 Activity Diagram 3 for Admin Deactivating Group

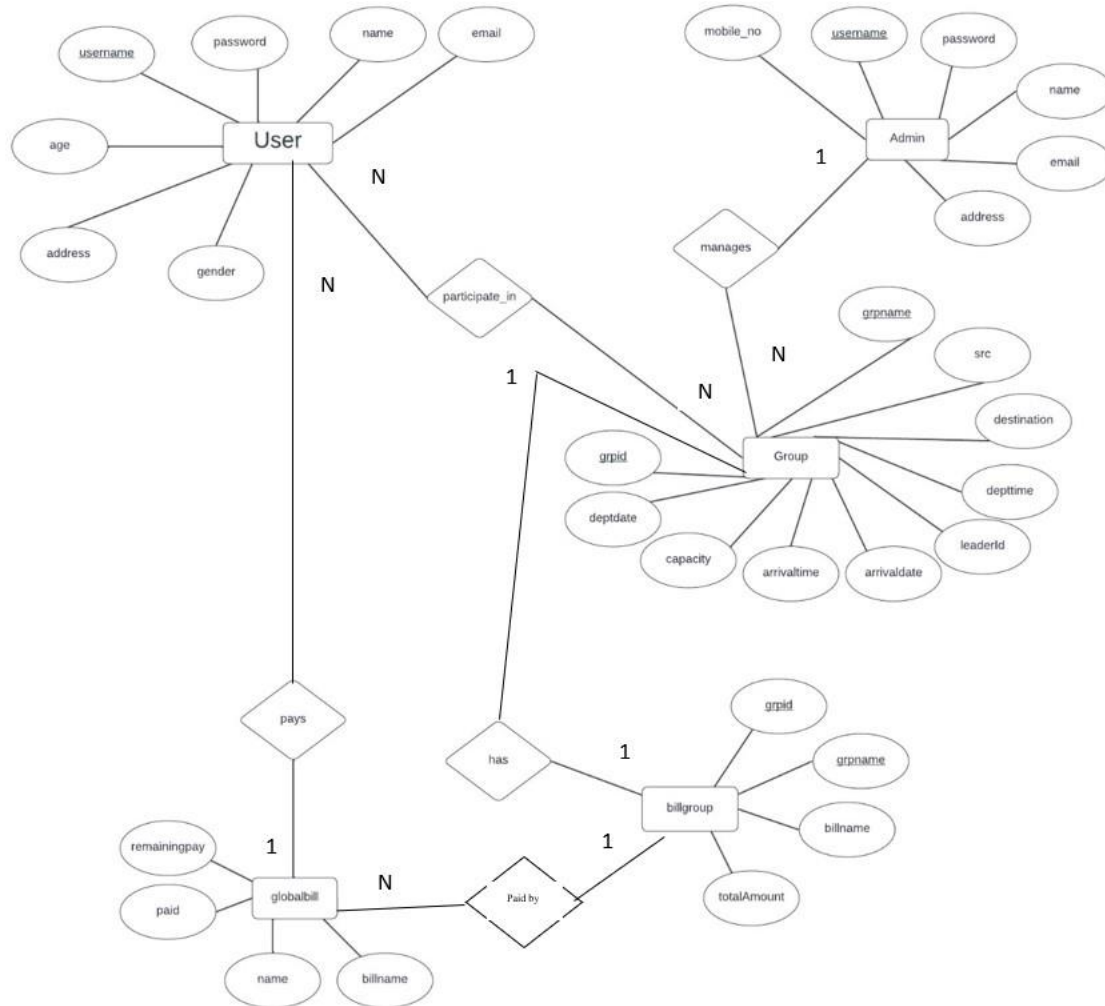
Activity Diagram showing the flow of activities performed for deactivating the group after the group has reached the destination. Admin can view all the groups and the group which has already reached the destination can be removed by the admin.



**Activity diagram for Admin**

## 2. Database Design

### 2.1 ER Diagram



**ER DIAGRAM**

## 3. Implementation Plans

### 3.1 Technology Stack

- Java Swings : **Swing** is a GUI widget toolkit for Java. It is part of Oracle's Java Foundation Classes (JFC) – an API for providing a graphical user interface (GUI) for Java programs.

Swing was developed to provide a more sophisticated set of GUI components than the earlier Abstract Window Toolkit (AWT). Swing provides a look and feel that emulates the look and feel of several platforms, and also supports a pluggable look and feel that allows applications to have a look and feel unrelated to the underlying platform. It has more powerful and flexible components than AWT.

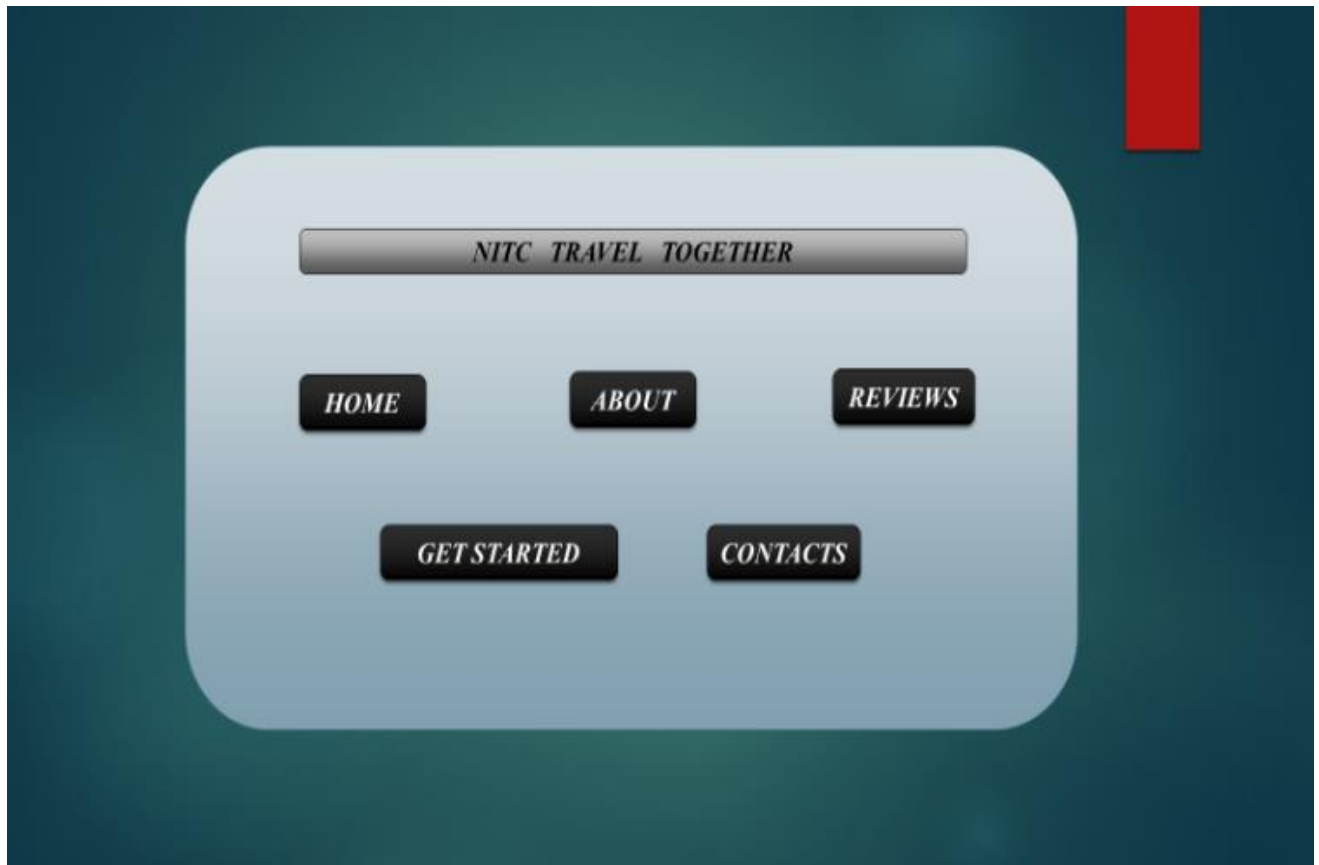
- Java : Java is a **programming language** and a **platform**. Java is a high level, robust, object-oriented and secure programming language.

Java was developed by *Sun Microsystems* (which is now the subsidiary of Oracle) in the year 1995. *James Gosling* is known as the father of Java. Before Java, its name was *Oak*. Since Oak was already a registered company, so James Gosling and his team changed the name from Oak to Java.

- SQL : Structured Query Language, abbreviated as SQL S-Q-L, sometimes "sequel" for historical reasons), is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream management system (RDSMS). It is particularly useful in handling structured data, i.e. data incorporating relations among entities and variables.

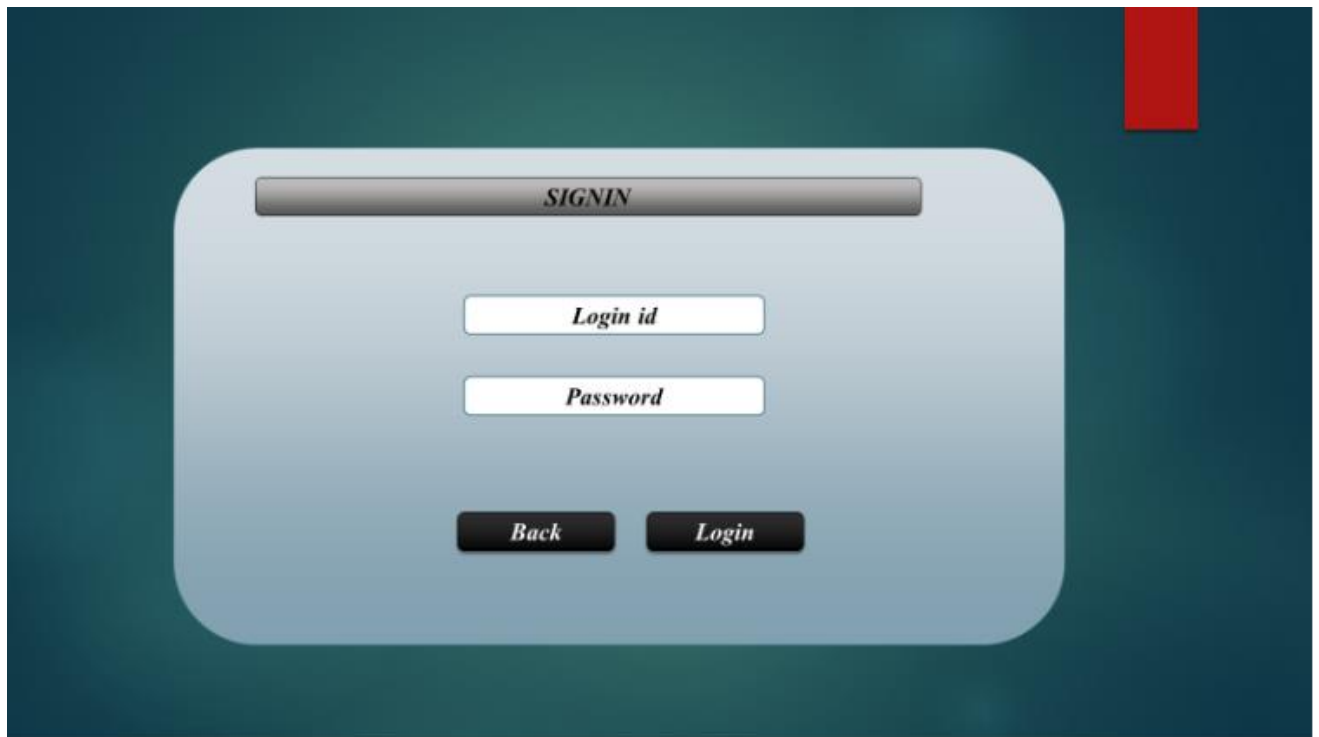
### 3.2 User Interface Prototyping

**Interface 1:** It is the home page where a user have multiple options to get the information, user can click on 'get started' to login or register in the application, a user can also see the reviews by clicking on 'review' button and a user is allowed to contact us by clicking on 'contact' button, a user can read some data regarding our application by clicking the button 'about'.



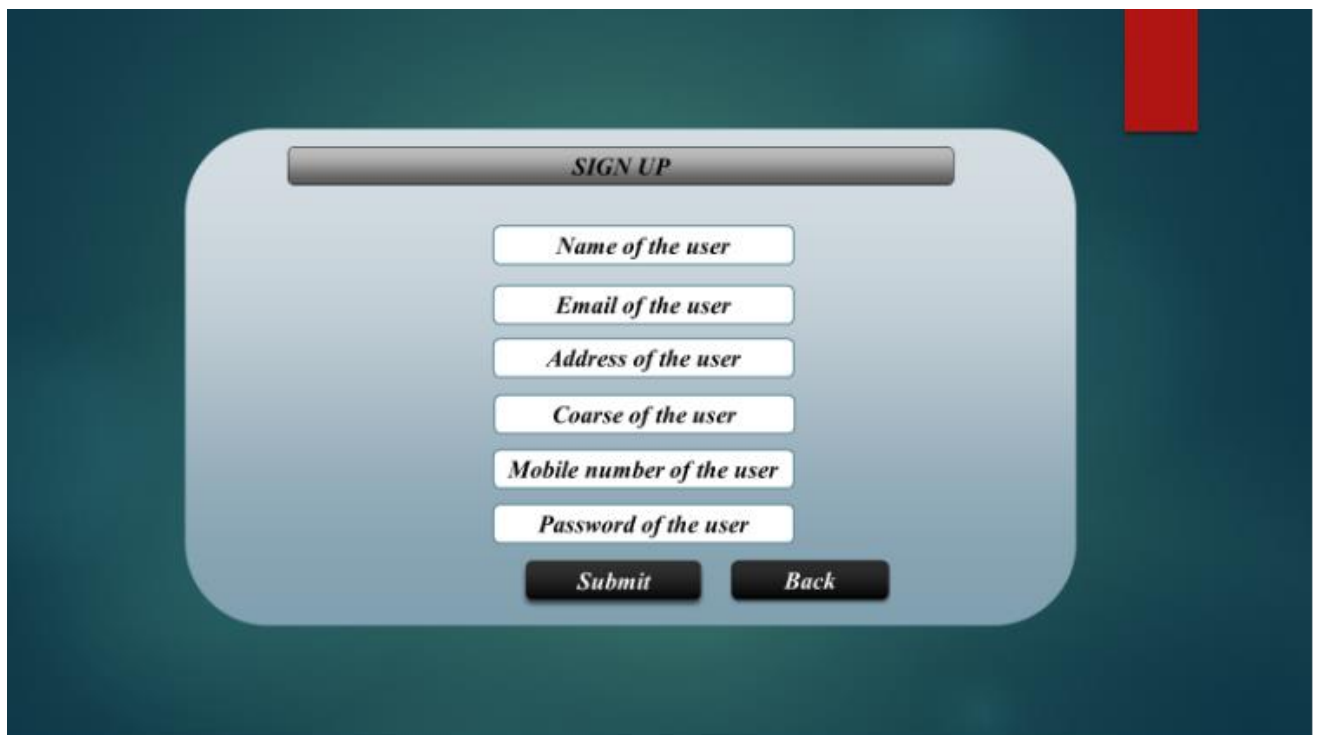
**Interface 2:** This is the sign in page where a user can login in our application. A user is able to sign in if he/she is already registered in our application, otherwise he/she is not allowed to login into the application. If the user is not allowed to login into our application he/she has to register in our application. If he/she already registers then the user has to type the user id which is the email id and the password that the user has created at the time of registration and then the user has to click login to enter into the system.





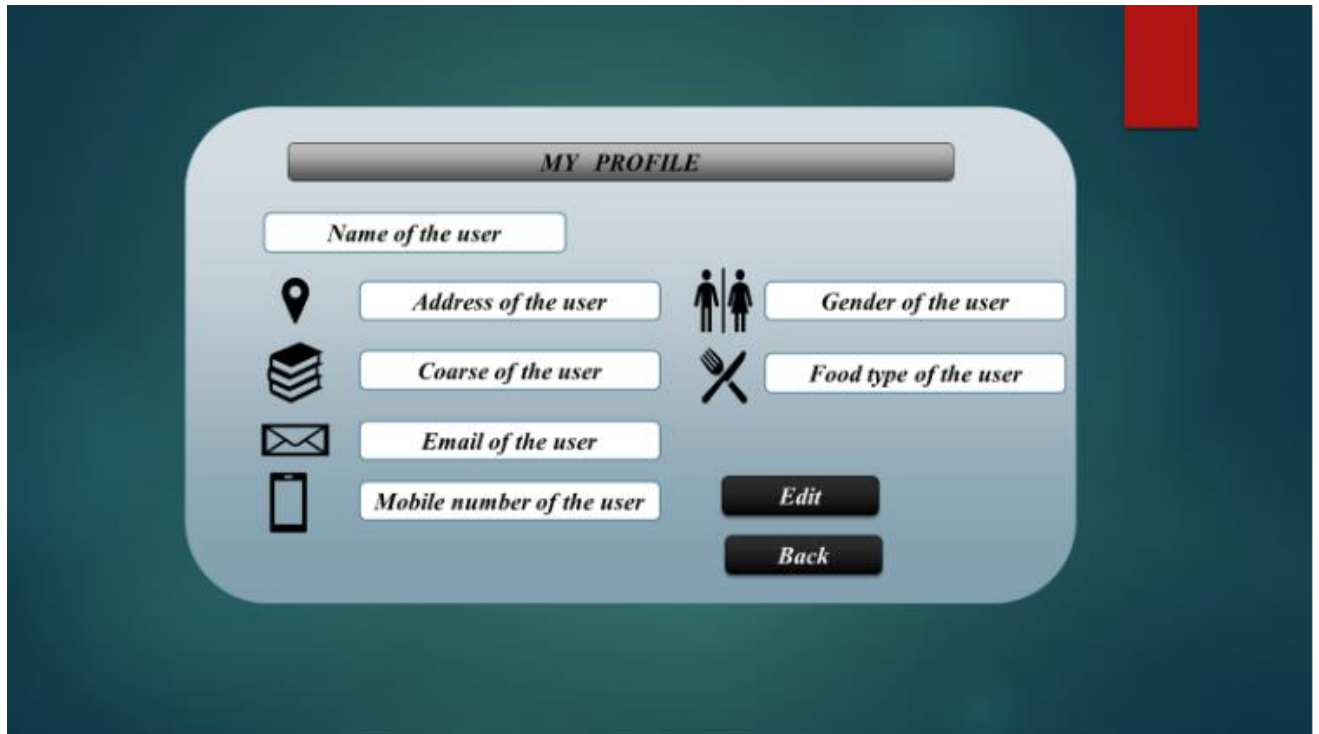
The image shows a 'SIGNIN' interface. It features a light blue rounded rectangle centered on a dark teal background. At the top of the rectangle is a dark grey bar with the word 'SIGNIN' in white. Below this are two white input fields: the first is labeled 'Login id' and the second is labeled 'Password'. At the bottom of the rectangle are two dark grey buttons: 'Back' on the left and 'Login' on the right. A red vertical bar is visible on the right side of the dark teal background.

**Interface 3:** This is the signup page which helps users to register in our application. Users have to give some credentials like name ,email id,address,coarse,etc. and then press the submit button to register in our application .



The image shows a 'SIGN UP' interface. It features a light blue rounded rectangle centered on a dark teal background. At the top of the rectangle is a dark grey bar with the words 'SIGN UP' in white. Below this are six white input fields stacked vertically, labeled: 'Name of the user', 'Email of the user', 'Address of the user', 'Coarse of the user', 'Mobile number of the user', and 'Password of the user'. At the bottom of the rectangle are two dark grey buttons: 'Submit' on the left and 'Back' on the right. A red vertical bar is visible on the right side of the dark teal background.

**Interface 4:** This is the my profile page where the user can see his profile and he can traverse his information like email, education and all such things and can edit his details by clicking the edit button.



The screenshot displays a user profile interface titled "MY PROFILE". It features a light blue rounded rectangle on a dark teal background. The profile fields are arranged in two columns. The left column includes icons for a location pin, a stack of books, an envelope, and a mobile phone, each next to a text input field. The right column includes icons for a man and a woman, and a fork and knife, each next to a text input field. At the bottom right of the profile card are two dark buttons labeled "Edit" and "Back".

MY PROFILE	
Name of the user	
Address of the user	Gender of the user
Coarse of the user	Food type of the user
Email of the user	
Mobile number of the user	
Edit	
Back	

**Interface 5:** This is the create group page where a user can become the admin of his group by creating his group by giving some information regarding his group like size, total fare and destination etc.



The screenshot displays a "CREATE GROUP" interface. It features a light blue rounded rectangle on a dark teal background. The form contains five text input fields stacked vertically, each with a label. At the bottom are two dark buttons labeled "Back" and "Create".

CREATE GROUP
From where journey ends
From where journey starts
Total number of persons
Means of transport
Total fare of the group
Back
Create

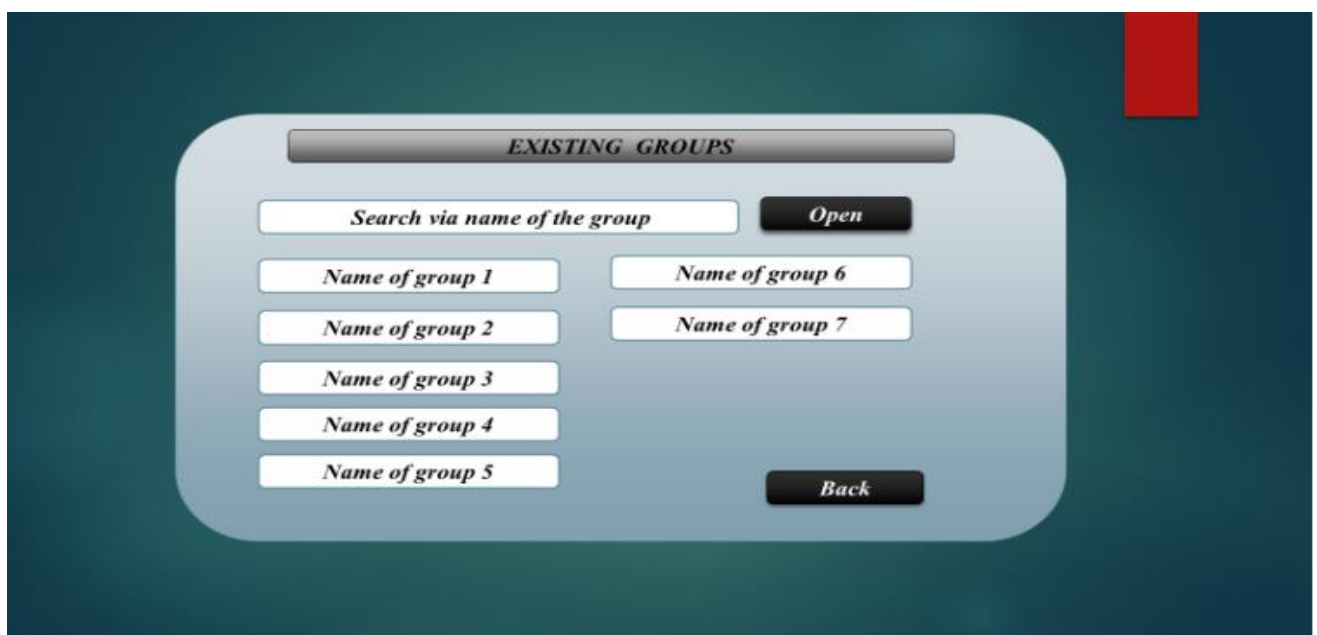
**Interface 6:** This is the group details page where the user can see the details of the group he created earlier, like total member and destination and total fare.

The user can add in the group and also opt out from the group .



The screenshot shows a mobile application interface titled "GROUP DETAILS". It features a light blue rounded rectangle on a dark teal background. Inside the rectangle, there are several input fields and buttons. The input fields are labeled: "Name of the admin", "Size of the group", "From where journey starts", "From where journey ends", "Means of transport", and "Total fare of the group". To the right of these fields are buttons labeled "Split bills", "pay", "Travel alert", "Meal", and "Entertainment". At the bottom of the rectangle are three buttons: "Back", "Add", and "Optout". A red vertical bar is visible on the right side of the screen.

**Interface 7:** This is the existing groups page where the user can see all the groups and the user can open any group via the group name and then the user can add into the group and also opt out from the group.



The screenshot shows a mobile application interface titled "EXISTING GROUPS". It features a light blue rounded rectangle on a dark teal background. Inside the rectangle, there is a search bar labeled "Search via name of the group" and an "Open" button. Below the search bar, there are seven input fields labeled "Name of group 1" through "Name of group 7". At the bottom right of the rectangle is a "Back" button. A red vertical bar is visible on the right side of the screen.

**Interface 8:** This is the admin panel page where admin can remove the group from the existing group, as the group leaves the starting location.



The screenshot displays a web interface titled "ADMIN PANEL" within a light blue rounded rectangle. The panel is set against a dark teal background. At the top right of the teal area, there is a small red rectangular element. The "ADMIN PANEL" title is centered in a grey bar. Below the title, there are two input fields: "Name of the group" and "Time when the group leave". At the bottom of the panel, there are two buttons: "Remove" and "Logout".

## 4. Test Cases

### 4.1 Test Case #1 (User Registration TC1)

**Author:** Karan Sachinkumar Satish

**Test Case Description:** Registering on the Application in order to use it.

**Pre-Conditions:** User should have valid mobile number and an NITC email id.

**Test Steps:**

Enter name

Enter Email

Enter Address

Enter Course

Enter Mobile Number

Enter Password

**Test Data:**

Karan Sachinkumar

[karan\\_m210687ca@nitc.ac.in](mailto:karan_m210687ca@nitc.ac.in)

Mumbai, Maharashtra

MCA

9702026749

abcd@nitc

**Expected Result:**

Successful Registration of the User

**Post Condition:**

Users must not create more than two id's in order to make it a spamless environment.

## 4.2 Test Case #2 (User Login TC2)

**Author:** Karan Sachinkumar Satish

**Test Case Description:** Logging in the Application in order to use it.

**Pre-Conditions:** User should have Registered before Login

**Test Steps:**

Enter Email

Enter Password

**Test Data:**

[karan\\_m210687ca@nitc.ac.in](mailto:karan_m210687ca@nitc.ac.in)

abcd@nitc

**Expected Result:** Successful Login.

**Post Condition:** Users Can start using the Application

## 4.3 Test Case #3 (User Registration TC3)

**Author:** Karan Sachinkumar Satish

**Test Case Description:** Editing the already created profile.

**Pre-Conditions:** User should have valid mobile number and an NITC email id and should have registered.

**Test Steps:**

Go to My Profile page and click on Edit button.

Then modify the detail you want to change and click save.

**Test Data:**

Name : Karan Sachinkumar Satish

**Expected Result:**

Name Successfully changed.

**Post Condition:**

None.

**4.4 Test Case #4 (Create Group TC4)**

**Author:** Atul kumar Sethi

**Test Case Description:** Creating new Group to be visible to other users.

**Pre-Conditions:** User should be Login before creating group.

**Test Steps:**

Enter Starting/Boarding Point

Enter Destination

Enter Number of Person

Enter Means of Transport

Enter Fare

**Test Data:**

kattangal

Wayanad

5

Bus

2000

**Expected Result:**

Successful Group Created.

**Post Condition:**

Group visible to other members and they can join that group.

**4.5 Test Case #5 (Create Group TC5)**

**Author:** Shivam kumar Jha

**Test Case Description:** Joining already Created Group .

**Pre-Conditions:** User should be Login before joining group.

**Test Steps:**

Login  
Go to Existing group Page  
Select the group/Search the group by groupname  
Click on Join Group

**Test Data:**

GroupName : WayanadVisitors

**Expected Result:**

Successful Group Joined.

**Post Condition:**

Group visible to other members along with increased count of already joined persons.

#### **4.6 Test Case #6 (Optout from group TC6)**

**Author:** Atul Kumar Sethi

**Test Case Description:** User already joined the group can opt out through following steps.

**Pre-Conditions:** User should be logged in and added to one of the existing groups.

**Test Steps:**

Select Group / Enter Group name  
Click OptOut

**Test Data:**

Group Name : e.g. WayanadTravellers(Supposing there is a already a group named WayanadTravellers )



**Expected Result:** Successfully Opted out and necessary database changes take place.

**Post Condition:** None

#### 4.7 Test Case #7 (Deleting the group TC7)

**Author:** Shivam kumar jha

**Test Case Description:** After Successfully reaching the destination admin can delete the group.

**Pre-Conditions:** Admin should be logged in and the group should have reached their destination.

**Test Steps:**

Select Group / Enter Group name

Click Remove

**Test Data:**

Group Name : e.g. WayanadTravellers(Supposing there is already a group named WayanadTravellers )

**Expected Result:** Group Successfully removed and necessary database changes take place.

**Post Condition:** None

## 5. Traceability Matrix

	<i>Design Elements</i>				
<i>Requirements</i>		<i>Class Diagram</i>	<i>ER Diagram</i>	<i>Activity Diagram</i>	<i>Sequence Diagram</i>
<i>Register</i>	<i>Yes</i>			<i>Yes</i>	<i>Yes</i>
<i>Login</i>	<i>Yes</i>			<i>Yes</i>	
<i>Create Group</i>	<i>Yes</i>	<i>Yes</i>	<i>Yes</i>	<i>Yes</i>	<i>Yes</i>
<i>Join Group</i>			<i>Yes</i>	<i>Yes</i>	<i>Yes</i>
<i>Deactivate Group</i>	<i>Yes</i>			<i>Yes</i>	<i>Yes</i>

## References

- <https://www.javatpoint.com/java-swing/>
- <https://www.youtube.com/>
- Lucid visual collaboration suite: Log in. (n.d.). <https://lucid.app/users/login>
- Lucid visual collaboration suite: Log in. (n.d.). <https://lucid.app/users/login>
- ActivityDiagram YouTube: <https://www.youtube.com/watch?v=CDPjTYSnCt4>
- Sequence Diagram YouTube :  
<https://www.youtube.com/watch?v=QQcVNjGVcGg>