CSE2004 (EPJ)

DATABASE MANAGEMENT SYSTEMS

FINAL PROJECT SUBMISSION

Project on:

VITMUN Application

SUBMITTED BY: **Team 36**

MEMBERS:

Khyati Chaturvedi (19BCI0124)

Parul Tripathi (19BCI0147)

School of Computer Science and Engineering,
Vellore Institute of Technology,
Vellore, Tamil Nadu, India – 632014

HELP SECTION

Software Requirements

Front End: Android Studio 4.1 (Kotlin)

Database: SQLite

Connection String:

```
class SQHelper (context: Context): SQLiteOpenHelper (context,
SQHelper.DB_name, null, SQHelper.DB_VERSION)
```

Database Details

- No. of tables as per the normalized schema: 12
- No. of tables in the final project : 12

Front end details:

How many interface pages? : 6

Type of interface (web page/application) : Application

Installation and Testing

Installing Android Studio on Windows:

- Download an .exe file (recommended), and launch it.
- Alternate: Download a .zip file, unpack the ZIP, copy the android-studio folder into the Program Files folder, and then open the android-studio > bin folder and launch studio64.exe (for 64-bit machines) or studio.exe (for 32-bit machines).
- Follow the setup wizard in Android Studio and install any SDK packages that it recommends.

Import the project in android studio. After, that:
Open Android Studio and if any project is open, close it (Go to File->Close Project):

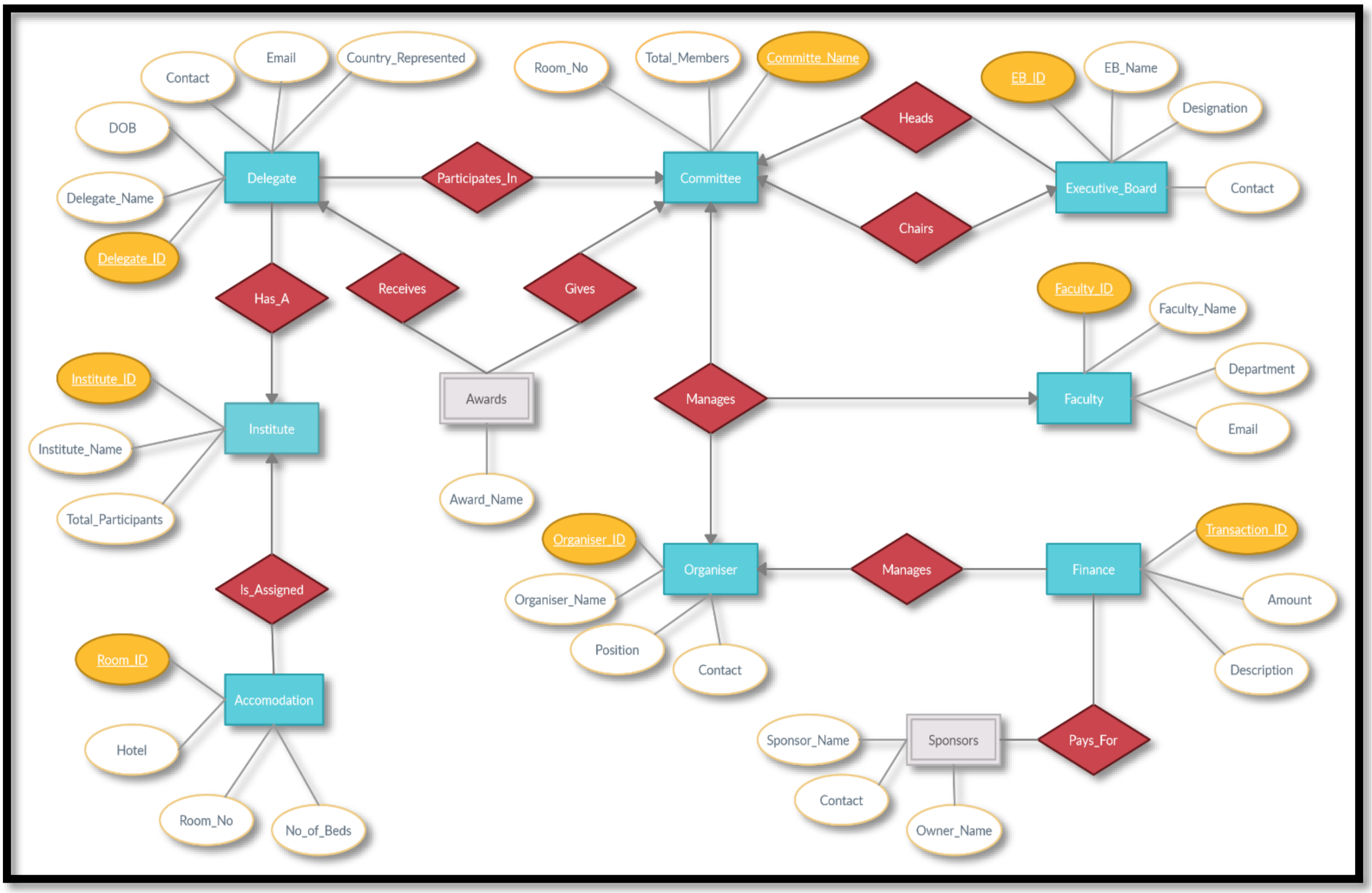
- **Step 1: Open Android Studio project:**
  - Click on "Open an existing Android Studio project" to open Android Studio project
  - Select your project directory from the 'Open file to Project' dialogue and click on the OK button.
  - Wait until project sync and build
- **Step 2: Open Gradle project:**
  - Click on “Import Project(Gradle)”
  - Browse the project by navigating the location where the project is stored; the project’s folder will appear with an Android logo on it.
  - Select the app and click OK, this will take some time for the Gradle to build (10-12 minutes).

  - An error might occur here stating the different SDK location.
  - In case of the above error just go to Project Directory and then find the file named “local.properties” in the root folder of the project. Open this file and scroll to the last 2 lines “ndk” and “SDK” and change the location with the one of the SDKs and save the file.
  - Again open the project or else if already opened in Android studio go to Gradle → Rebuild.
- **Step 3: To build and run the app:**
  - In the toolbar, select the app from the run configurations drop-down menu.
  - From the target device drop-down menu, select a device to run the app on.
  - **Note:** If there are no devices configured, then we need to either connect a device via USB or create an AVD to use the Android Emulator.
  - Click on Run button in Toolbar or Select the Run menu in the menu bar.
- **Step 4: Edit project:**
  - By default, Android Studio displays the project files in the Android view. This view does not reflect the actual file hierarchy on disk, but is organized by modules and file types to simplify navigation between key source files of the project, hiding certain files or directories that are not commonly used. Some of the structural changes compared to the structure on disk include the following:
  - Shows all the project's build-related configuration files in a top-level Gradle Script group.
  - Shows all manifest files for each module in a module-level group (when there are different manifest files for different product flavours and build types).
  - Shows all alternative resource files in a single group, instead of in separate folders per resource qualifier. All density versions of the launcher icon are visible side-by-side.
- **Within each Android app module, files are shown in the following groups:**
  - manifests: Contains the AndroidManifest.xml file.
  - java: Contains the Java source code files, separated by package names, including JUnit test code.
  - res: Contains all non-code resources, such as XML layouts, UI strings, and bitmap images, divided into corresponding sub-directories.
  - Gradle Scripts: Two types of Gradle files are used in android projects; one is Project level Gradle which contains application repositories, dependencies, and project-level variables, and the second type are modules level Gradle which contain app version name, version code, min version, application ID, dependencies for this module, debug and release build types, build flavours, etc.

Phase I: Requirement Analysis

1. **Delegate:** Delegate_ID, Delegate_Name, DOB, Contact, Email, Country_Represented
2. **Committee:** Committee_Name, Total_Members, Room_No
3. **Institute:** Institute_ID, Institute_Name, Total_Participants
4. **Accommodation:** Room_ID, Hotel, Room_No, No_of_Beds
5. **Organiser:** Organiser_ID, Organiser_Name, Position, Contact
6. **Executive_Board:** EB_ID, EB_Name, Designation, Contact
7. **Faculty:** Faculty_ID, Faculty_Name, Department, Email
8. **Sponsors:** Sponsor_Name, Owner_Name, Contact
9. **Finance:** Transaction_ID, Amount, Description
10. **Awards:** Award_Name

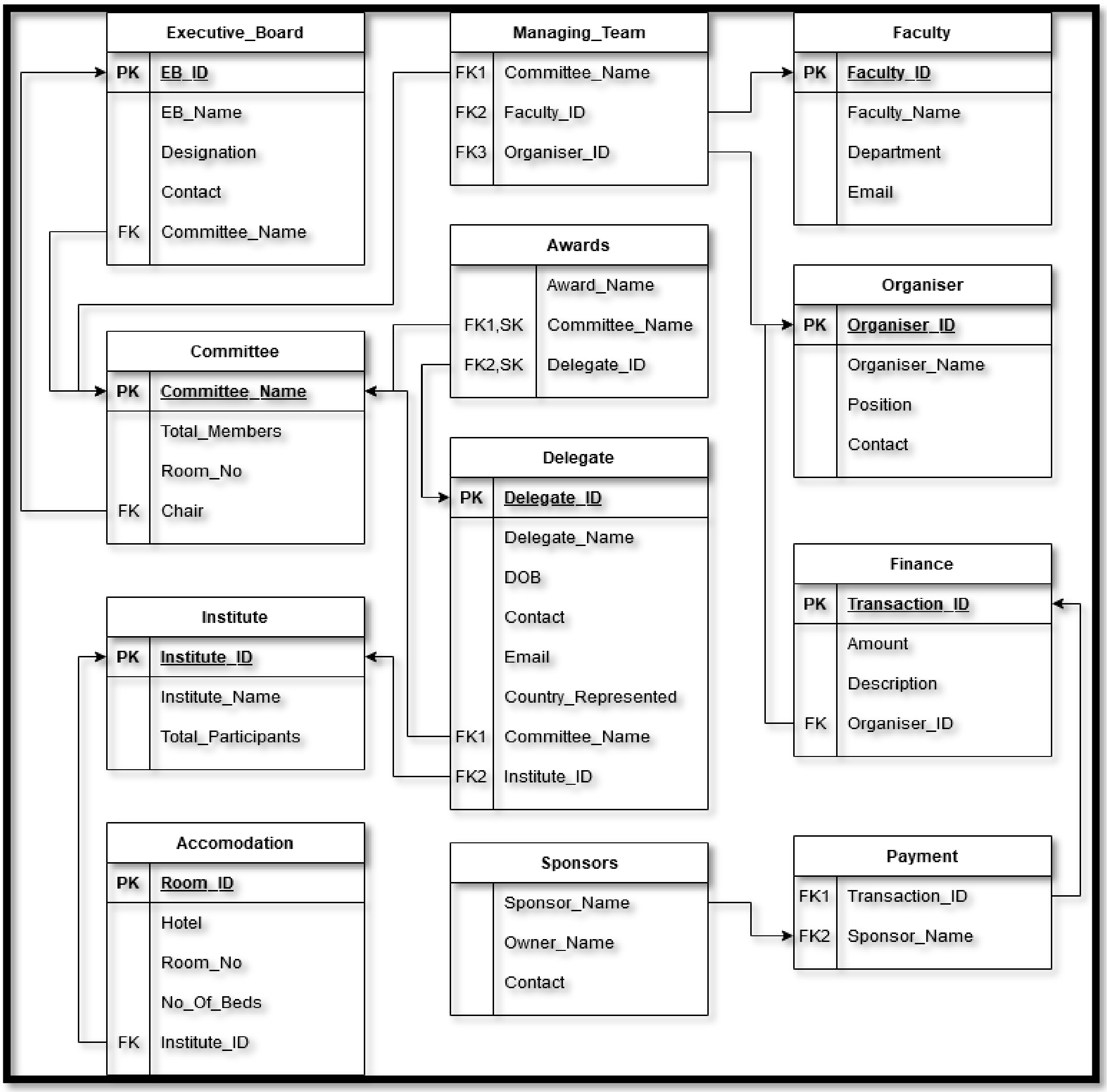
ER Diagram



- Sponsor and Awards are weak entity sets.
- All attributes are simple and single valued.
- **One-to-one relations:** Executive_Board-*Chairs*-Committee
- **Many-to-many relations:** Sponsor-*Pays_For*-Finance
- Rest all are one-to-many relations.
- Committee, Faculty and Organiser are in a ternary relation.
- All relationship sets are strong.

Schemas

- *Bold underlined – primary key; Double underlined – foreign key*
- **Delegate:** Delegate_ID, Delegate_Name, DOB, Contact, Email, Country_Represented, Committee_Name, Institute_ID
- **Committee:** Committee_Name, Total_Members, Room_No, Chair
- **Institute:** Institute_ID, Institute_Name, Total_Participants
- **Accommodation:** Room_ID, Hotel, Room_No, No_of_Beds, Institute_ID
- **Organiser:** Organiser_ID, Organiser_Name, Position, Contact
- **Executive_Board:** EB_ID, EB_Name, Designation, Contact, Committee_Name
- **Faculty:** Faculty_ID, Faculty_Name, Department, Email
- **Managing_Team:** Committee_Name, Faculty_ID, Organiser_ID
- **Sponsors:** Sponsor_Name, Owner_Name, Contact
- **Finance:** Transaction_ID, Amount, Description, Organiser_ID
- **Payment:** Transaction_ID, Sponsor_Name
- **Awards:** Award_Name, Delegate_ID, Committee_Name



Phase II - Normalization

Final Schemas

- *Bold underlined – primary key; Double underlined – foreign key*
- **Delegate:** Delegate_ID, Delegate_Name, DOB, Institute_ID
- **Delegate_Position:** Delegate_ID, Committee_Name, Country_Represented
- **Delegate_Contact:** Delegate_ID, Contact, Email
- **Committee:** Committee_Name, Total_Members, Room_No, Chair
- **Institute:** Institute_ID, Institute_Name, Total_Participants
- **Accommodation:** Room_ID, Hotel, Room_No, No_of_Beds, Institute_ID
- **Organiser:** Organiser_ID, Organiser_Name, Position, Contact
- **Executive_Board:** EB_ID, EB_Name, Designation, Contact, Committee_Name
- **Faculty:** Faculty_ID, Faculty_Name, Department, Email
- **Managing_Teams:** Committee_Name, Organiser_ID, Faculty_ID
- **Sponsors:** Sponsor_Name, Owner_Name, Contact
- **Finance:** Transaction_ID, Amount, Description, Organiser_ID
- **Payment:** Transaction_ID, Sponsor_Name
- **Awards:** Award_Name, Delegate_ID, Committee_Name

After normalization, the database has 14 tables.

Phase – III

Hardware and Software Requirements

Frontend: Android Studio 4.1

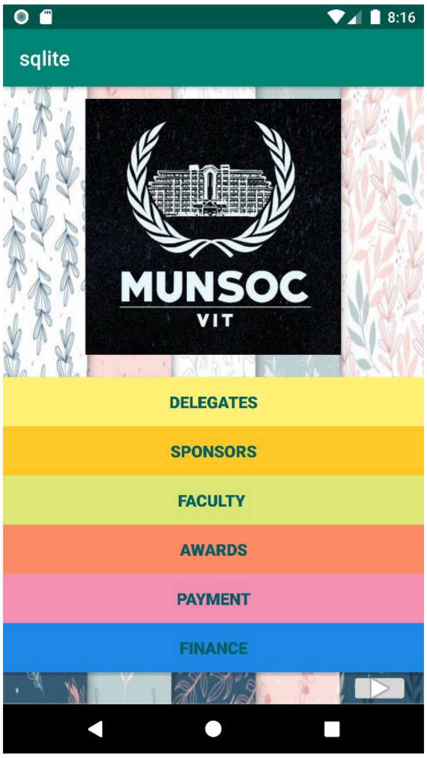
- Microsoft® Windows® 7/8/10 (64-bit)
- 4 GB RAM minimum, 8 GB RAM recommended.
- 2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution.

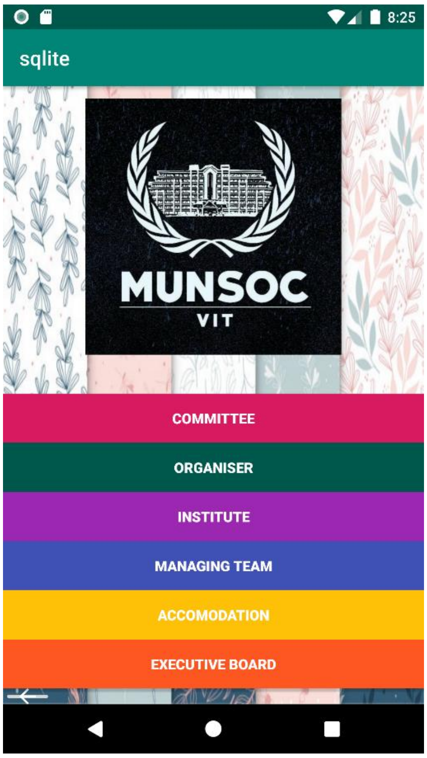
Backend: SQLite

- XP Service Pack 3/2003 Service Pack 2 or higher.
- . NET Framework 2.0 or higher.
- 500 MB RAM required, 1+ GB recommended.
- Adequate free disk space for transmission logging and writing received files.

Flow of Control

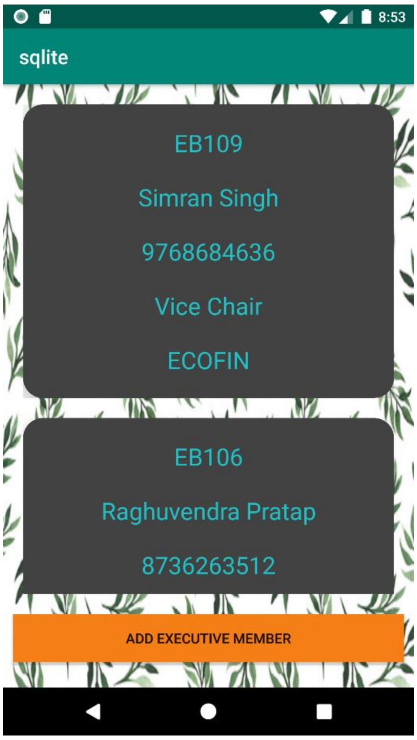
1. ***First screen:*** Here, we can see the schemas. On the first screen, we see half the schemas, and on clicking on the right arrow on the bottom right part of the first screen, we see the remaining schemas on the second screen.





Screen 1 **Screen 2**

2. ***Navigating the application:*** On clicking any button, we can go to the respective, entity set, and add, view and search (Awards and Managing Team) data. Here, we have opened Executive Board.



EB109

Simran Singh

9768684636

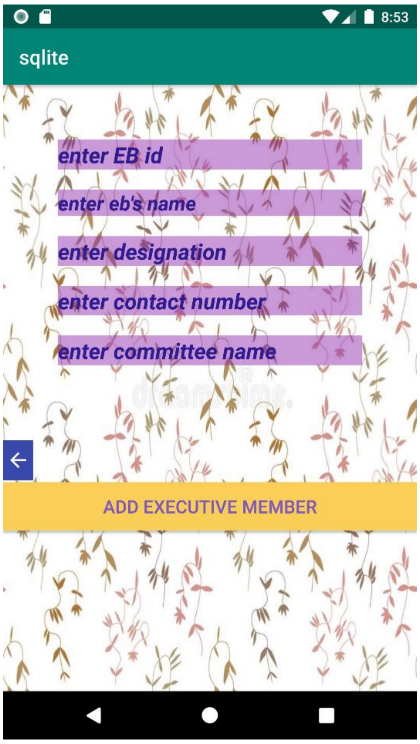
Vice Chair

ECOFIN

EB106

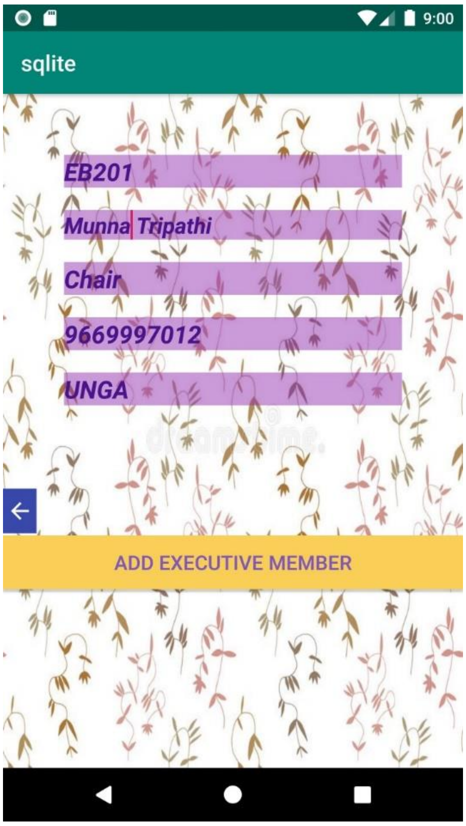
Raghuvendra Pratap

8736263512



Viewing the table **Adding data**

3. ***Adding Data:*** Here, we have entered a record in the table Executive Board. Data can be added in any table in this way. Every table has its own addition pane.



9:00

sqlite

EB201

Munna Tripathi

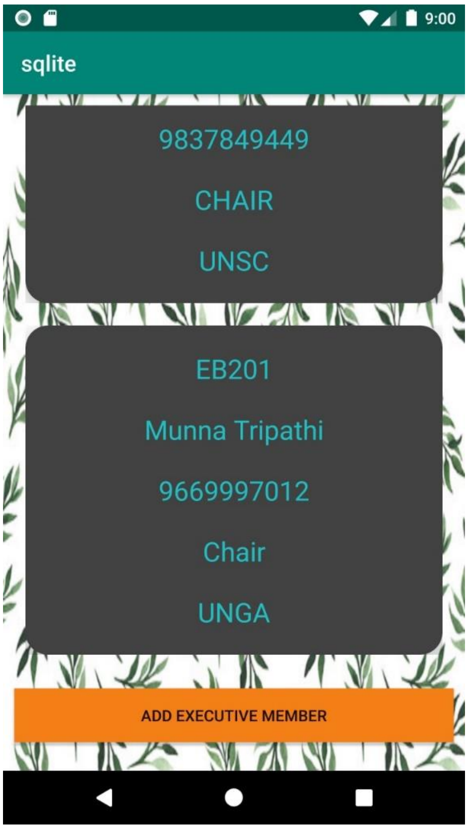
Chair

9669997012

UNGA

ADD EXECUTIVE MEMBER

Entering data



9:00

sqlite

9837849449

CHAIR

UNSC

EB201

Munna Tripathi

9669997012

Chair

UNGA

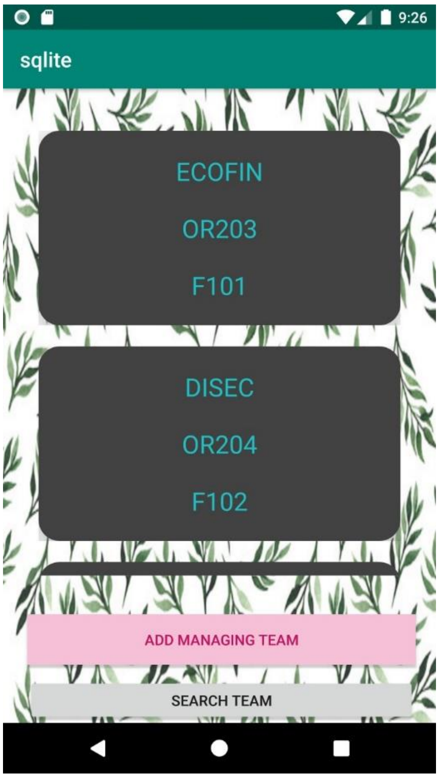
ADD EXECUTIVE MEMBER

Displaying entered data

We return to the main screen (Screen 1) using the back button. Then we can open any other table.

4. ***Searching using a specific attribute:*** This option is available for Awards and Managing Team. We can see all the records at once, and can also filter the records on the basis of a particular attribute. We have a back button on the search sections.

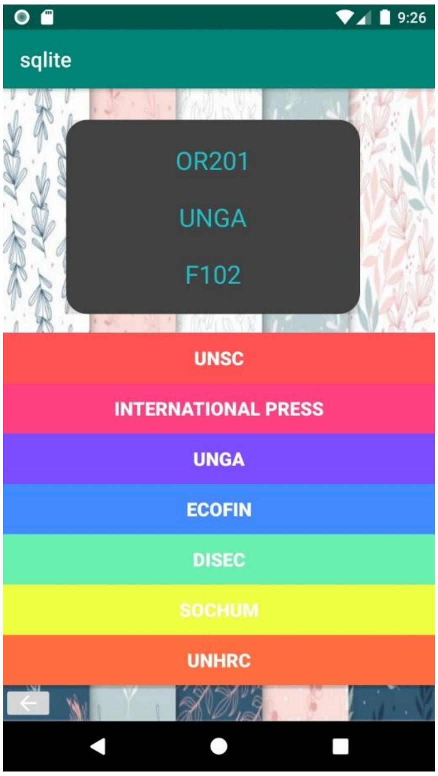
For Managing Team:



ECOFIN
OR203
F101

DISEC
OR204
F102

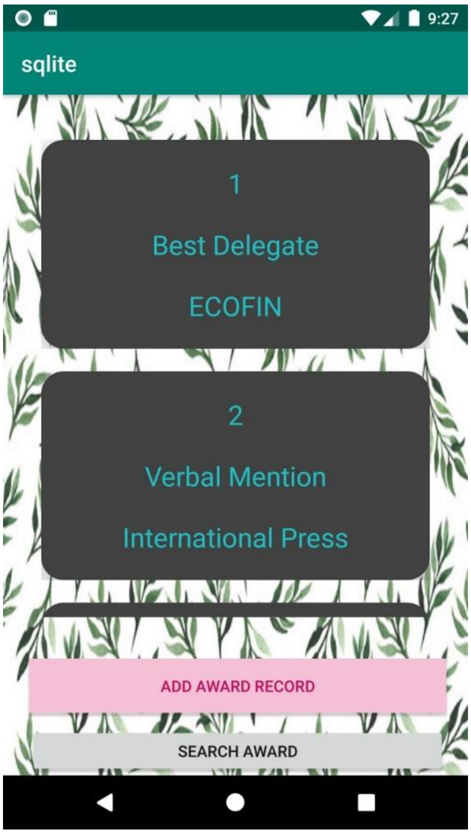
Entire table



OR201
UNGA
F102

Searching by committee

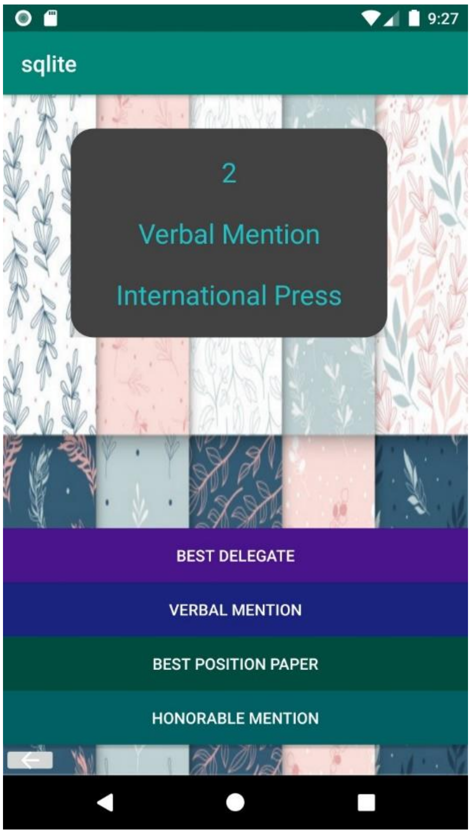
For Awards:



1
Best Delegate
ECOFIN

2
Verbal Mention
International Press

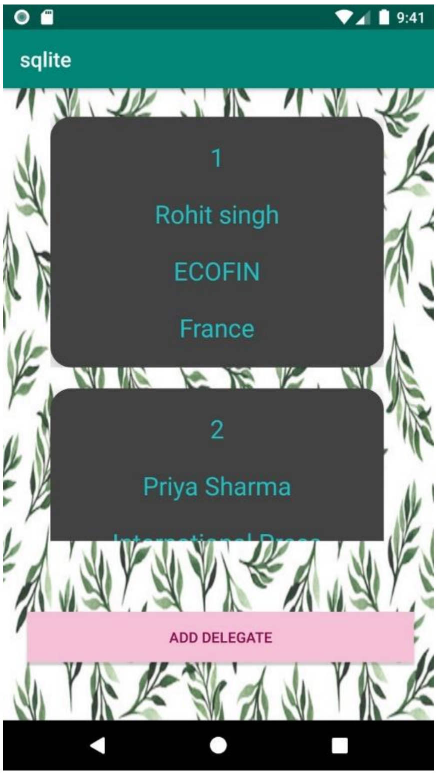
Entire table



2
Verbal Mention
International Press

Searching by Award Name

Project Gallery



1

Rohit singh

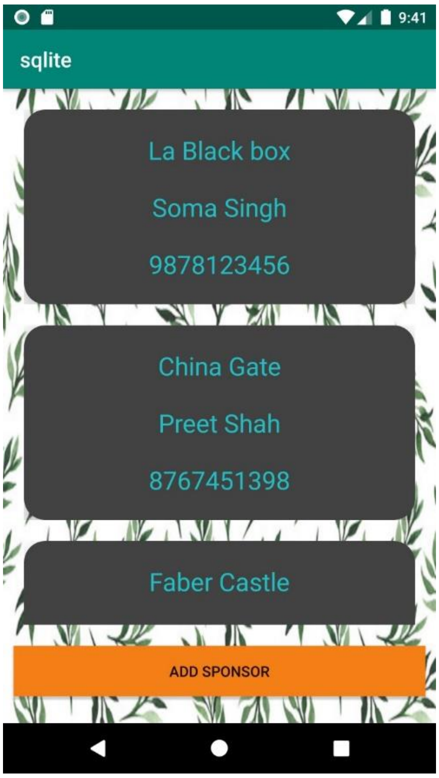
ECOFIN

France

2

Priya Sharma

Delegate



La Black box

Soma Singh

9878123456

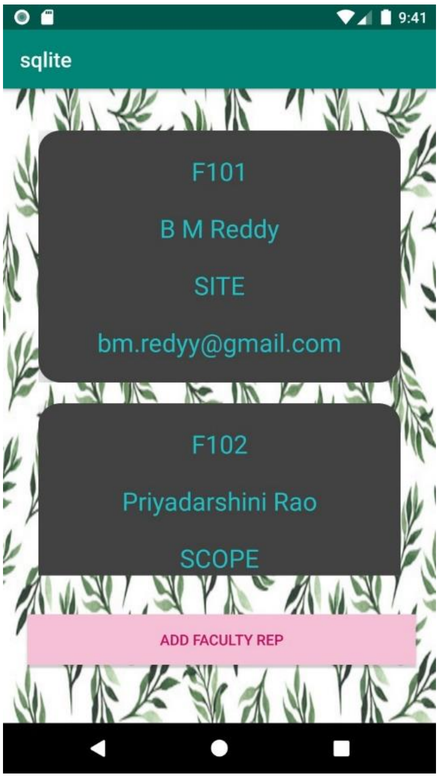
China Gate

Preet Shah

8767451398

Faber Castle

Sponsor



F101

B M Reddy

SITE

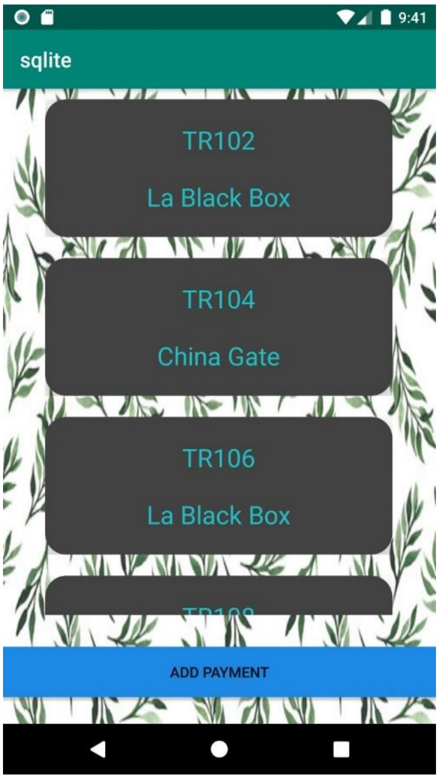
bm.redyy@gmail.com

F102

Priyadarshini Rao

SCOPE

Faculty Rep



TR102

La Black Box

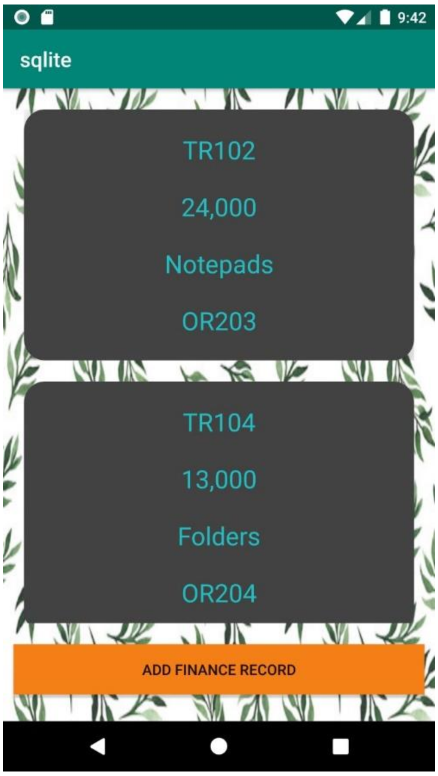
TR104

China Gate

TR106

La Black Box

Payments



TR102

24,000

Notepads

OR203

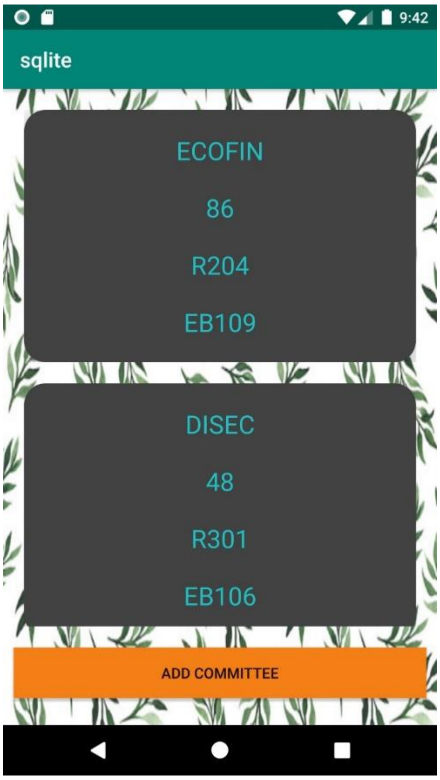
TR104

13,000

Folders

OR204

Finances



ECOFIN

86

R204

EB109

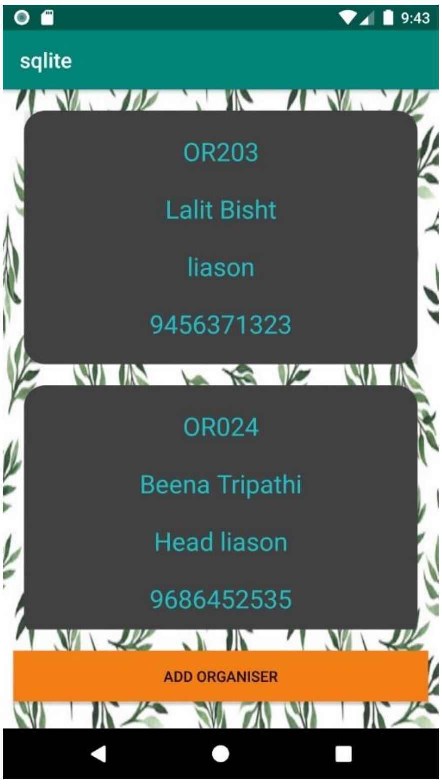
DISEC

48

R301

EB106

Committees

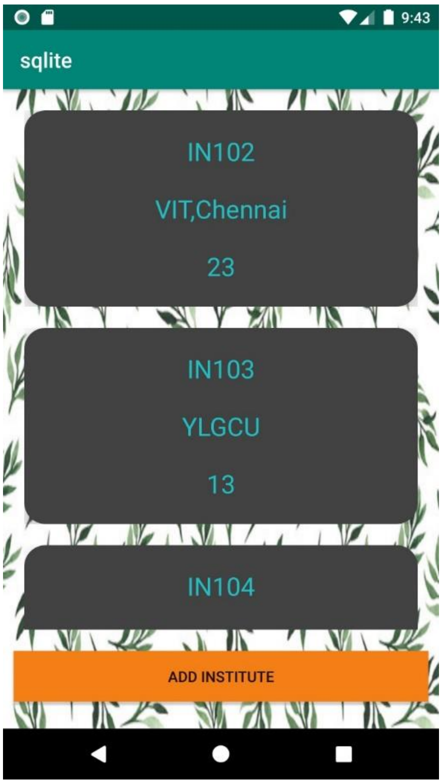


sqlite

OR203
Lalit Bisht
liason
9456371323

OR024
Beena Tripathi
Head liason
9686452535

Organiser



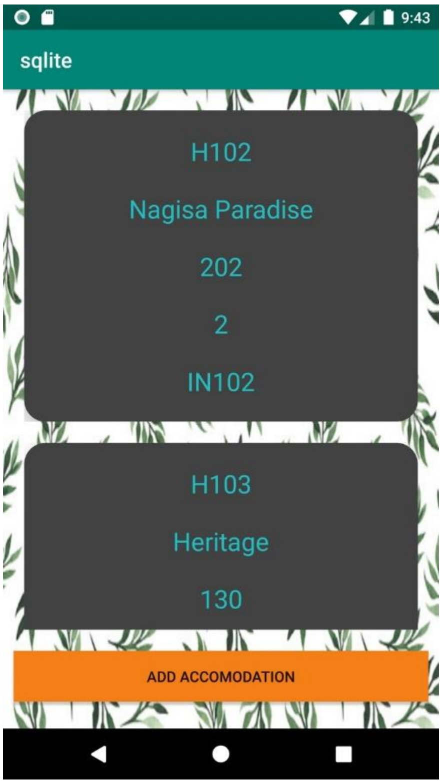
sqlite

IN102
VIT,Chennai
23

IN103
YLGCU
13

IN104

Institute



sqlite

H102
Nagisa Paradise
202
2
IN102

H103
Heritage
130

Accommodation

Learning Product and Experience

The ideation process and brain storming to form a concept behind our application was the first step and it was only after we reached the database formation process we realized that we need an actual concept to have a meaningful database.

This project helped us realize the thorough process of database formation and normalization when one develops backend. Through various phases under the guidance of **Prof. Saravanakumar K**, we were able to understand the working behind full stack development as we connect frontend and backend of an application. We learned Kotlin as a part of developing our frontend and came to understand UI/UX designing and its importance. This was a fruitful learning experience which made a significant difference in our skills from before.