Abstract

An Android application which acts as a platform for a user to stay updated about the different activities/events happening near him/her. The user tells the app about his interests and based on them will get feeds about the events happening. Also there is a group-chat interface where users with same interest can interact and discuss. There is a separate event-maker feature especially for the coordinators. The main feature of this app is that groups are created in genealogical order. For example, if the student is of Hostel 2, of Cse Dept., pursuing M.Tech., this app provides options to chat among all the hostel 2 members and as the info matched increases the group size reduces, so a vast variety of groups are created using this idea. Initially the target users will be the people in institute, so the app acts as a medium for people with similar hobbies to plan events(via group-chat), get notification of the events(via events feed) and thus stay connected within insti via virtual cable; InstiCable.

1 Introduction

Our Purpose is to provide a platform where users can see events and programs they are interested in. Often Event Co-ordinators find it hard to penetrate event’s knowledge to the interested students due to improper communication channels and lack of response from the students. We believe, InstiCable can solve this by Generating personalised feeds based upon the user’s interest so as to avoid cluttering of unnecessary events, at the same time helping Co-ordinators to contact the ‘relevant’ Junta. We also implemented a "Group Chat" feature so people having similar interests(e.g. Cricket) , can chat in a Dedicated ‘Cricket’ group, this will help them to call on valuable players which otherwise would be a headache( Going Room by Room, Asking whether they wish to Play is O(Rooms) , difficult for a human to process).

2 Software used

- Android Studio - Free software.
- Firebase for backend and database.
- Latex and Doxygen for documentation.
- Git to effectively merge and manage work of each member.

3 Hardware required

As it is a real time app, so it needs to fetch data from the server every time a user logs in to update him about the new events and chats. Therefore, the device has to be connected to internet for this app to function.
4 User Interface and Implementation

4.1 How to Sign Up?

Sign In must be a simple process and so it is.
User Enters their e-mail, InstiCable checks whether the User Already Exists, if Yes, it takes them directly to their personalized Feed.
Otherwise it asks their Name and Password.

InstiCable then asks your Interests and Details. It Uses these information as key-words to generate your feed, you can change these details later in the Settings as well.
InstiCable then asks your interests to show relevant events only.

5 Events Page

After user clicks on Go button in interests page, the events are fetched from the server meeting his interests. Each event is displayed in CardView having date, time, venue and description fields. Events are stored in Recycler View which highly optimizes the overhead of storing all the views in a layout and rather holds the views using adapter and displays them when scrolled down. It has an extra feature for coordinators as they can create an event here and push it to the the database. For all of this pushing and fetching, firebase is used. Database is created in firebase once events are pushed as JSON objects and are fetched later using listener functions attached to the events in real-time.
6 Notifications

The most expected feature which is required in any app is its capacity to notify users of the activities happening related to the app while he/she is not using it. To enable notifications in our app i.e. notify the user of the events happening of his/her interests, we have used the messaging and function services available in the Firebase for backend management and Android’s notification layout for the front end.

7 Like or Dislike Event

It displays the number of people interested in this event, thus enabling coordinator to know how many people will be there for this event helping him plan out resources.

8 Group Chat

Group chat is also implemented using groups created based on number of interests matched. A user can take part in group chat by clicking on the group icon on the topmost bar. Groups are created in genealogical order, implies there are subgroups within a group where number of interests matched increases as user go further down into the group.
9 User Profile and Settings

There is a menu option for Settings, clicking on which user sees their profile and can change their preferences like Hostel, Department etc.
10 What we did and what we proposed?

We Initially Proposed the Following Feature Set:

1. Generating Events based upon user’s interests (done).
2. Settings page for changing one’s preferences
3. Separate activities for coordinators as they only have the ability to create an event
4. Implementing Group Chat in Genealogical framework (done).
5. Functionality where user can decide whether they are interested in going an event, we proposed that the Statistics thus obtained would be sent to Event Co-ordinators (done).
6. Notifications regarding new event pushed of his interest (done)

So, we effectively did all that we proposed.

11 Limitations

This app is suitable to a small section of people as it generates the groups automatically. Also, at present there is no special privilege for a user to be coordinator. If a coordinator selects multiple interest groups, multiple notifications come, we were unable to debug it.

12 References