

RAMAIAH INSTITUTE OF TECHNOLOGY
(Autonomous Institute Affiliated to VTU)



Department of Information Science and Engineering

Jan - May 2017

A Report On
“Group Chatting Application”
Submitted in partial fulfillment for CodeSprint of the subject
Applied CS With Android
by

Agasthya HD 1MS14IS141

Shubham Pawar 1MS15IS418

Under the guidance of
Mr. Shashidhara HS
Associate Professor
Department of ISE, RIT

Internal Examiners	External Examiner
Mr. Shashidhara HS Associate Professor Dept of ISE, RIT	

INDEX

Content	Page No.
1.Introduction	1
2.Platform and Compatibility	1
3.Tools Used	2
4.Result and Description	3
5.DataStructure used	4
6.Activities and Methods	5
7.Screen shots of UI	6-8
8.Other examples of Datastructures used.	9
9.References	10
6.Conclusion and Summary	11

INTRODUCTION

The chatting android application is a modern and useful application which allows signed-in users to chat with other users of the app publicly and also share photos among other users of the app. The signing-in can be done through google account or an other email account. Once logged the user can see all the messages and photos sent by other user. The user can sign-out from the app once he has finished reading.

We have used the firebase-UI Api , firebase-storage Api and firebase-database Api, firebase-messaging, firebase-auth api to handle various functioning of the app.

PLATFORM AND COMPATIBILITY

The minimum android version should be 4.0(Ice Cream Sandwich) .

API Level >14 or above is required.For some features API like Firebase-UI require Api-18.

TOOLS USED

1.Android Studio:

Android Studio is the official IDE for Android application development, based on IntelliJ IDEA. On top of the capabilities you expect from IntelliJ, Android Studio offers:

- Flexible Gradle-based build system
- Build variants and multiple apk file generation
- Code templates to help you build common app features
- Rich layout editor with support for drag and drop theme editing
- Lint tools to catch performance, usability, version compatibility, and other problems.

2.Firebase:

Firebase is a mobile and web application development platform. Firebase is made up of complementary features that developers can mix-and-match to fit their needs.

i.Firebase cloud messaging:

Firebase Cloud Messaging (FCM) is a cross-platform solution for messages and notifications for Android, iOS, and web applications, which currently can be used at no cost.

ii.Firebase Auth

Firebase Auth is a service that can authenticate users using only client-side code. It supports social login providers Facebook, GitHub, Twitter and Google.

iii.Realtime Database

Firebase provides a realtime database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud.

iv.Firebase Storage

Firebase Storage provides secure file uploads and downloads for Firebase apps, regardless of network quality. The developer can use it to store images, audio, video, or other user-generated content. Firebase Storage is backed by Google Cloud Storage

RESULTS AND DESCRIPTION

USER INTERFACE

Firebase-UI is used as it simplifies the signing-in process and mechanism. At the firebase website we can add any type of sign-in methods like google, email account, facebook, github, twitter account. Also we can add features of email verification and password change. Empty message will not be expected from messenger. This once stored in firebase-database is used displayed to all users when they login-in or in a active session.

Firebase Database

The Firebase-database API is used to store all the app-users' text messages along with user-names and all the photos' url stored in firebase-storage. This is stored in terms of a Hash Map of key and values where unique message id is the key and values would name of the message text and url.

Firebase Storage

The firebase-storage is used to store all the images being sent by all the users which will be displayed to all users as soon as they login-in or they are in a active session. The url of this images will be sent to firebase database so that user's app can download the images along with sender's name as soon as it is sent.

Firebase RemoteConfig

The firebase remote-config allows us to update message-length and other configuration parameters from firebase console.

Handling the pause and resume state of the application:

When the user pauses the application, the authentication state listener are removed that is user is signed out and as soon as the app as the application resumes the authentication state listener are added again that is the user is signed-in again.

DATA STRUCTURE USED

The data structure used to store and retrieve the values in the firebase database is **HASH MAP**.

In computing, a hash table (hash map) is a data structure which implements an associative array abstract data type, a structure that can map keys to values. A hash table uses a hash function to compute an index into an array of buckets or slots, from which the desired value can be found.

In a well-dimensioned hash table, the average cost (number of instructions) for each lookup is independent of the number of elements stored in the table. Many hash table

designs also allow arbitrary insertions and deletions of key-value pairs, at constant average cost per operation.

The Hash Map is used primarily because **O(1)** for all its operations like get, put. In our project, this simplifies how data is stored in firebase as data is not structured like sql, no-schema to be implemented and all the implemenation is specified in the studio-code, no need to code in the console.

We have implemented such that each message has a key holds the unique identifier that identifies the key and value that holds the name and text or name or url of the image.

```
chatting-application-892ae
  messages
    -KeFTTrng17eqycg08nh
      name: "Agasthya Hd"
      text: "google"
    -KeFlrlINPrc27bhDFab
      name: "Agasthya Hd"
      text: "is the new chatting app"
    -KeJ34w0BJm0vC1q9CP2
      name: "Vishwanath Harekal"
      text: "hello"
    -KeJJpZAEDxhm7IpD0_z
      name: "Vishwanath Harekal"
      photoUrl: "https://firebasestorage.googleapis.com/v0/b/cha...)"
    -KeJh0YhmyqWjtn0KFS4 + ×
```

ACTIVITIES AND METHODS:

The Main Activity of the Group-Chat Application handles the main functionality of logging in and putting the text to the database and choosing the photos from the gallery and uploading it to the database.

The MainActivity has ChildEventListener which updates the app whenever we a message is putforth in the app.

There are various methods used in android application.

Firstly, **getInstance()** method to get an instance/object of Firebase-database, Firebase-Auth and Firebase-RemoteConfig() and Firebase-storage.

The **getReference()** method is used to connect to particular HashMap structure/location stored in the database. The method is used to get reference of the Firebase-storage location.

Then, **onTextChanged()** method of the addTextChangedListener makes sure that blank text message cannot be sent by enabling the send button only when text typed. The setfilters method is used to set default text length.

The main method used to upload messages to the database is the **push()** method. Custom messages can be pushed using the push().setValue().

OnAuthStateChanged() is method that gets the username from FirebaseAuth and implements other methods OnSignedIntialize() and OnSignedInCleanUp()

The **onActivityResult()** method handles the Sign-In , upload and download of photo from the database.

The addAuthenticationStateListener and removeAuthenticationStateListener methods adds and removes the AuthenticationStateListener object when the application goes to a pause and resumes to ensure that app is signed-out when app is paused and not in an active session.

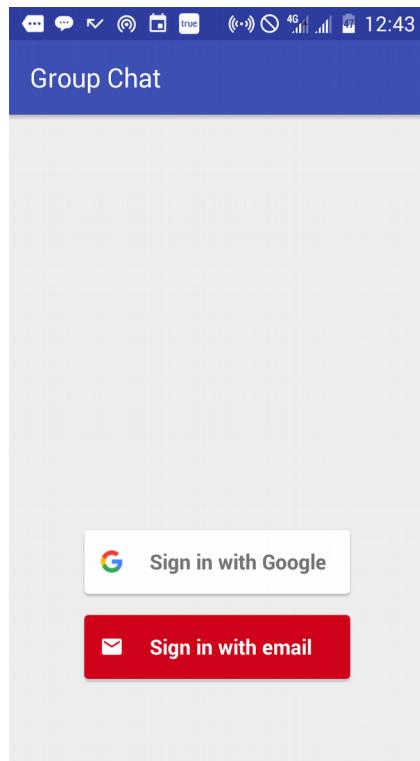
The **fetchConfig()** method is used to fetch any configuration from the firebase-console(firebase website).

The **attachDatabaseListener()** and **detachDatabaseListener()** are used to create and destroy the **childEventlistener** which has various methods that listen to the database and specifically we use **onChildAdded()** method that gets invoked whenever we add an item to the database.

We have used the Glide image loading and cache library to load images based on url and used appropriate placeholder to place the images.

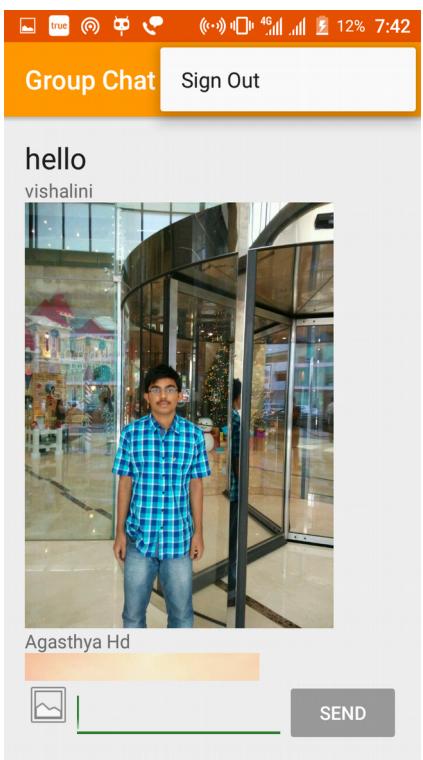
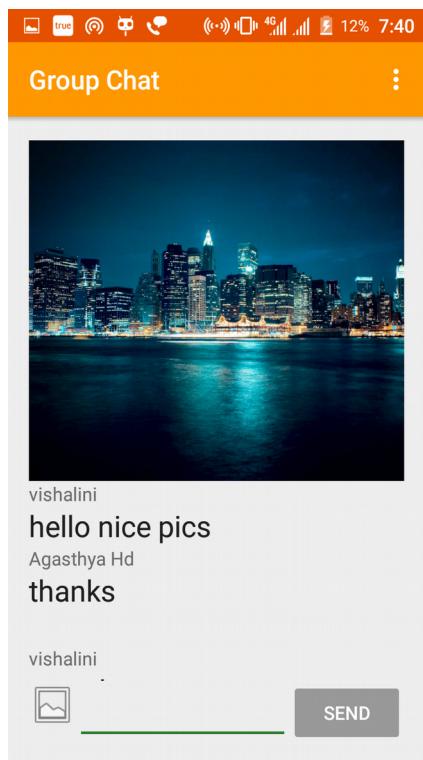
SCREEN SHOTS

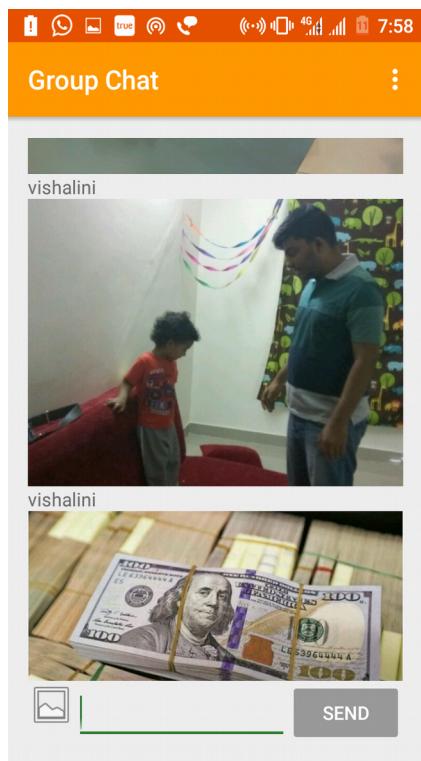
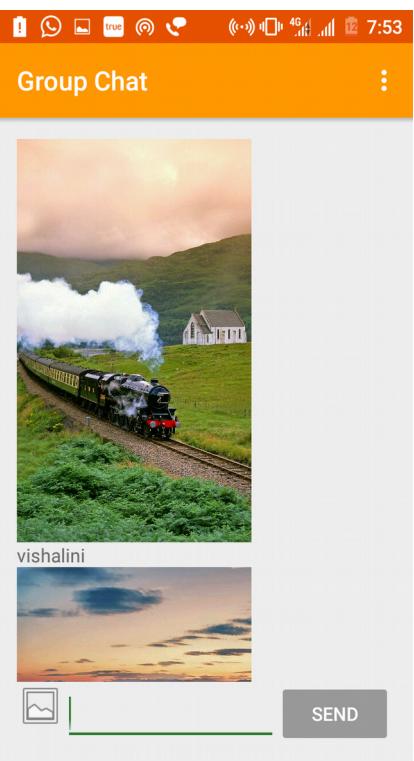
LOGIN PAGE



CHATTING PAGE:







OTHER EXAMPLES OF DATA STRUCTURES USED:

The application has used customised ArrayList to make the List view of the app to display the chat text sent and the photos shared.

Hash Map, the data structure primarily used in the application has wide variety of applications like telephone directory to store names and numbers, with names as key and corresponding details of the person as value.

REFERENCES:

Refererred to **developer.android.com**, the website meant for Android App Development by maintained by Google.

Referred to Udacity website which has various courses for Android App Development **in.udacity.com**

CONCLUSION AND SUMMARY:

The group chat application is a simple application that handles the real time messaging and image sharing seamleously.The application is a prototype of the full-scale mesagging and sharing android application.More features like firebase-analytics to analyse app user behavior and patterns and firebase-notification to notify desired set of users can be implemented.The UI can be added more features to make more user-friendly than ever.