CAMPFIRE ANDROID APPLICATION

MINOR PROJECT I

Submitted by:

Paras Lehana (9914103234)
Pradeep Asthana (9914103254)
Rajnish Upadhyay (9914103208)

Under the supervision of:
Mr. Sudhanshu Kulshrestha

Department of CSE/IT
Jaypee Institute of Information Technology University, Noida

DECEMBER 2016

ACKNOWLEDGEMENT

I would like to place on record my deep sense of gratitude to Mr. Sudhanshu Kulshrestha, CSE Faculty, Jaypee Institute of Information Technology, India for his/her generous guidance, help and useful suggestions.

I express my sincere gratitude to Mr. Gaurav Saxena, Dept. of Computer Science & Engineering, for her stimulating guidance, continuous encouragement and supervision throughout the course of present work.

I also wish to extend my thanks to Dr. Mukesh Saraswat and other classmates for their insightful comments and constructive suggestions to improve the quality of this project work.

Signature(s) of Students

Paras Lehana (9914103234)

Pradeep Asthana (9914103254)

Rajnish Upadhyay (9914103208)

ABSTRACT

Accounting for more than half of the presently used hand-held devices, Android, as an operating system, has provided users with great opportunity to innovate and get things done in a mobile device. It is because IT has become the new part of the world. There is a big need of different applications. People want software for every specific task to make the work easier. We have developed the android application on "Social Connections and Groups".

This android app helps you find and join your real-life groups online whether you are a Company employee or just a college student, Campfire will find existing groups based on your query as well as allow you to create a new one. It is not like a simple social and groups app but it is a feature-full professional social environment where you have different tabs to interact with people, have personal chats, share documents and media, post announcements, search forums, have discussions or just anything which makes it virtual environment of your real-life groups.

This Project allowed me to increase my knowledge in Java, a language with which we had many difficulties, but also to discover different aspects of privileges and authority to a Social Networking Android Application.

GANTT CHART

REQUIREMENT GATHERING	■						
ANALYSIS & DESIGN		■	■				
CODING			■	■	■	■	■
TESTING				■	■	■	■
VALIDATION & IMPLEMENTATION					■	■	■
DOCUMENTATION					■	■	■
DATE	**AUG 23, 2016**	**SEP 12, 2016**	**SEP 25, 2016**	**OCT 7, 2016**	**NOV 15, 2016**	**NOV 30, 2016**	**DEC 3, 2016**

LIST OF FIGURES

ABBREVIATIONS & NOMECLATURE

• **Android:** Developed by Google, a popular operating system for smart phones.

• **Java:** Java is a programming language and computing platform first released by Sun
Microsystems in 1995. There are lots of applications and websites that will not work unless you have Java installed, and more are created every day. Java is fast, secure, and reliable. From laptops to datacentre, game consoles to scientific supercomputers, cell phones to the Internet, Java is everywhere!

• **IDE:** a programming environment integrated into a software application that provides a GUI builder, a text or code editor, a compiler and/or interpreter and a debugger.

• **SRS:** Software requirement Specification.

• **OS:** Operating System. It is an interface between user and hardware.

• **OAuth:** OAuth is an authentication method where you approve applications to act on your behalf without sharing your password.

• **FCM:** A Firebase Cloud Messaging (FCM) Android client is a client app that runs on an Android device.

• **Volley:** Volley is an HTTP library that makes networking for Android apps easier and most importantly, faster.

• **Firebase:** Firebase is a mobile and web application platform with tools and infrastructure designed to help developers build high-quality apps. Firebase is made up of complementary features that developers can mix-and-match to fit their needs.

• **JSON:** JavaScript Object Notation (JSON) is a minimal, readable format for structuring data. It is used primarily to transmit data between a server and web application, as an alternative to XML.

- **XML:** Extensible Mark-up Language (XML) is a mark-up language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

TABLE OF CONTENTS

Page No.

1. INTRODUCTION

1.1 DOCUMENT PURPOSE

The purpose of this document is to present a detailed description of Campfire which is a Social Networking based Android Application. It will explain the purpose and features of the application, the interfaces of the application, what this application will do, the constraints under which it must operate and how the application will react to external stimuli. This document is intended to be evaluated by the mentor and the panel members for this minor project.

1.2 SCOPE OF PROJECT

The main objective of our project is to provide real life roles of the users in their organization through profile badges and designation in the Organization. Hierarchy levels are properly defined and verified smartly through email and admin panel. Groups are highly manageable and customizable. This Android Application is open for and accessible to all platforms whether it's a corporate group or just a small level institute like government school.

Any user can register on the app and choose any group either from existing or create a new one. The user will be able to access all the features like chatting, profile view and media access. Verified Groups are just another feature of the app. Users can access and use the app even without verifying their designations and groups. However, their designations, profile and groups would not show a verified badge unless they are verified by us. This makes our app accessible to everyone as well as keeping authenticity and verification still integrated into the app.

Keeping groups verification by our side, user will be able to choose among the Verified Groups of every organization. Groups can verify themselves by email and any existing world organization can request the verification. Supporting Digital India initiative, this can help all small level organizations and institute of the country to get on the internet world.

Not every institute or organization have web portals or intra-networks for organization connectivity, thus, Campfire can help them get online and connect with its users.

Thus, the scope of the application extends to every internet user as anyone is either associated to a group or can join our free rooms. However, the application is mainly dedicated to organization connectivity rather than personal networking. Existing world Organization and Groups are the main entities of the scope of the project.

1.3 BACKGROUND STUDY

The Social Networking Apps are very much accepted and satisfying the normal Human being requirements. This Application helps us to communicate fast and vastly with others at very low cost than older way of communicating through SMS.

Mostly Application these times have the Group Chat features as well as private chat with feature of end -to-end Encryption technique to secure the shared information. After going through the various apps based on group chat, their real-time application. It helps us to thinking about the chat section of this app with some new and different features and makes it more users friendly and for some official private works.

Before starting our project, we referred some research papers and similar existing apps and technologies which gave us fair knowledge and idea and made us clear about the work which we would do try to complete with the best possible outputs.

The detailed summary of the references which we have referred are:

- Firebase can power your app’s backend, including data storage, user authentication, static hosting and more. Data in the firebase is stored as JSON and synchronized in real time to every connected client. When you build cross platform apps with Android, iOS and JavaScript SDKs, all of our clients share one firebase database and automatically receive updates with the newest data.

- Firebase Authentication makes user authorization secure and private. We are not storing user credentials like password on anywhere on our app server database and the app admins, by any means, can’t view these too.

- Apart from IP Addresses or web cookies, JSON Tokens are nowadays used for all modern applications to identify a UID (Unique User Identification) associated with every user. We are storing these tokens on our app server too to deliver FCM based messages to selected segment of users.

- OAuth 2.0 or Open Authorization is an open standard for token-based authentication and authorization on the Internet. It allows an end user's account information to be used by third-party services, such as Facebook, without exposing the user's password. We have implemented OAuth services with Firebase to enable users to login with their existing Google and Facebook Accounts and fetch user details from third-party services to save user’s time spent while registering for the first time

- Chatting Applications are based on real time data fetching and storing and the synchronization time is the most important key of this feature to be considered. We have implement Firebase Real-time Database that updates the chat screen using its background sync technology.
- There are already many apps on the market for group connections and social networking but most of them lacks user identity and hierarchy definitions and verification. We have read Research Papers on Identity & Access Management (IAM) from Google Patent US6460141-B1 and the IAM Program Plan from Havard University (references added).
- Our application database server stores different user privileges associated with their roles. This restricts and allows privileges like banning user, adding/removing users to different users based on their roles. Domains are verified by the admin of the app while users requesting to join a particular domain are notified to and verified by the verified domain owners if any.

2. REQUIREMENT ANALYSIS

2.1. SOFTWARE REQUIREMENTS

- Platform: Android 4.0.1 & Up
- Language: Java Runtime Environment
- Internet Connection: 2G sufficient

2.2. HARDWARE REQUIREMENTS

Minimum Requirement is as follows:

- 128 MB RAM required
    - Processor with speed 500 MHz
    - Internet Connectivity: The system must run over the internet, all the hardware shall require to connect internet will be hardware internet for the interface.

2.3. FUNCTIONAL REQUIREMENT

This section provides requirement overview of the system. Various functional Modules that can be implemented by the system will be:

2.3.1: Description

- The User details must get stored in the databases on the ‘signup’ button click.
- The Login Page Activity must appear when application launches for the first time.
- User must redirect to the Profile Activity after the login has been done.
- User is able to communicate with other users.
- User is able to sign out from the Dashboard.
- If user is removed from the domain by the admin, pop-up should appear on Dashboard page.
- Removed User should be signed out as soon as the alert is cancelled
- If user is banned from the Chat Room, alert should be displayed.

2.3.2: Registration

- If an existing domain with the same description is found on the app server, users should be added to the existing domain and notification should be send to the domain admin.
- If no domain is found, users should be able to create a new one.
- On domain registration, user should be able to define roles associated with the group and their hierarchy.
- If user wants to be the admin of a domain, he/she should be able to request the claim and further verification would be done by the App Team themselves.
- On registration, user should be redirected directly to the Dashboard without requiring to login again.
- Roles will be displayed in a Spinner if domain already exists.
- If desiring to create new domain, the designation provided previously should be added into the database
- Verification email should be sent to the registered email ids to activate the account.
- Domain admin, if any, would be notified about the new user.

2.3.3: Login & Dashboard

- Customer logins to the system by entering valid user id and password to communicate with their organization's members.
- Upon login request, user is authenticated with the Firebase server. If verified, profile page must appear and if not, a Toast should be displayed.
- On no Internet Connection Error, exception should be thrown and displayed as an error.
- Upon user is authenticated by Firebase, Profile details is fetched from App Server in JSON format.
- Verification Badges are shown over domain name and user display picture if they are verified.
- Upon clicking verification, a toast is displayed with the short description.

- User is made accessible to different features namely Group Chat, Group View and Logout.
- Domain verification email is sent manually using pre-defined templates to the email address of concerned authority.

2.3.4: Logout

- After their desired activities, user can successfully log out from the Dashboard.

2.3.5: Group Chat

- If user is banned by the admin to participate or view Chat Room, an alert is displayed upon clicking the button. Dismissing it is the only option.
- Designations are displayed with the user names.
- On clicking a user name, his/her profile is displayed.
- Enter key is used to send messages that are stored on Firebase Real-time Database and displayed with Firebase Synchronization adapter variable.
- On any data changed on the domain chat database, activity updates the chat displayed.
- Emoji can also be sent.

2.3.6: Group View

- All of, but logged in, users of same domain are displayed in the List
- Upon clicking any user item, member profile page is displayed.
- Member page provides features such as Message, Remove User and Ban User.

- Logged in user can message another user of the same domain.
- If user is higher in hierarchy compared with the member, privileges like Removing/Adding User and Ban/Unblock User are displayed.
- Database entry for Ban_Code is updated upon clicking the corresponding buttons.
- Member Ban_Code is used to display the text inside the button and assign its function
- Verification badges are displayed according to the Verification Status.
- On clicking verification badges, toast is displayed with short description.

2.4. NON-FUNCTIONAL REQUIREMENTS

2.4.1: Usability

- To redirect directly to Dashboard, the application must have a registered user.
- Users must be notified against every wrong entry in either the signup page or the login page.
- The Data entered in the fields must be stored into the server databases.
- User authentication references should be sent to Firebase server
- The Data must be fetched from the databases efficiently.
- MySQL Database error should be handled and displayed.
- Progress bars should be displayed while any online operation is going on.

- Privileges button should change according to the hierarchy level and ban code.
- JSON Exceptions should be thrown and displayed in a toast.
- Alert Boxes should not respond on touch outside its scope.

2.4.2: Form validation while registering new user

- All the seven fields namely Name, username, Password, Personal email, Organizational email, Organizational domain, Designation.
- Username should not contain space
- Password---minimum length 8 characters.
- Organizational email--@ must be present.
- User email while logging in should be checked with general email verification string.

2.4.3: Compatibility

- The Application must only run on Android devices, no other platform is supported as of now.
- Different screen sizes should be supported
- Application is developed with the SDK Version of ICS and tested on Nougat and Marshmallow. It should run on different OS Versions

2.4.4: Implementation

- Application will be implemented in Java for Android API.

3. DETAILED DESIGN

3.1. PROBLEM INTRODUCTION

The project is to make a user-friendly app which provides real life roles of the users in their organization according to their hierarchy level. The project can be useful for small organization like schools and colleges. Most social networking and group chatting applications out there in the market maintains no hierarchy levels on user privileges and roles. Anyone can create the group and be its admin even if he/she is not supposed to be given those privileges based on the hierarchy level of his/her designation in existing world organization. Also, members joining later in the already made groups can be on the higher hierarchy level in the organization but he may have no special privileges in the online group. Apps lack isolated tabs that can isolate off-topic chats, official discussions and announcement section. Groups are mostly made by one and then users are added into it but most IM apps lack searching existing groups so that users willing to join them can send a joining request to its admin. Real life roles and designation should be verified so that privileges can be granted correctly. Group Verification is needed for groups claiming to be verified so that verification badges can be displayed beside them which can further help users to find the official groups online. Only verified groups will maintain member hierarchy.

3.2. INNOVATIVE IDEAS OF PROJECTS

Although, being a Social Networking and Connection app, it still has distinguishing features from the mainstream and popular ones. Although any one can register on this app, rather than personal connections, but Groups, Institutes & Organization are in the main scope of this application. Apps based on group chat and social connections lack group hierarchy level with access management and its required verification. Anyone can make the groups and thereby, manage the group as admin. For real life organization, groups, clubs and institutes to have its official domain in an app, domain, user and designation verification is ought to be there. This can further be used to define hierarchy levels and identity and access management rules that can be applied to a segment of roles. Searching groups based on verification status can help users to decide among different groups of the same requirement available online as well as know about the official one. All the chat in one place can create confusion among the important ones. Users might only want to be notified about the important announcements only. Campfire has multiple isolated custom chat creation feature and tabs including off-topic chat, official discussions and important announcements that helps isolating different activities based on users' interest or requirement. Admin of the domains can define and manage hierarchy level and add users to their domain upon verification. 3-way verification is implemented into app for handling different types of verifications. Account is verified by personal email addresses using mail service of our app server. Users are added to a group upon the verification by the group admin via app or email. The newly created domain is verified for the first time manually by the team using pre-defined email templates that are sent to the email of concerned authority. Upon their approval, user is designated the post of managing admin of the group and all the further verification task is handled by the admin from then.

The table of comparison between the two of most popular social apps, WhatsApp & Facebook is given below:

SNO	FEATURES	WHATSAPP	FACEBOOK	CAMPFIRE

1	Group Verification	✕	✕	✓
2	Search Existing Groups	✕	✓	✓
3	User Verification	✕	✕	✓
4	Members Roles	✕	✕	✓
5	Organizational Hierarchy	✕	✕	✓
6	Media Upload	✓	✕	✓
7	Isolated Tabs	✕	✕	✓

3.3. PROJECT OBJECTIVE

- Make an Efficient Communication Portal for any type of Organization and Groups.
- Have the facility for Uploading Media files.
- Isolated tabs within each Group to manage different Communication Types.
- Different Tabs within Groups for Announcements, Notices,
- Media, Off-Topic Chat, Private Chats and Primary Chat-box.
- ‘Verified Group’ Labels that helps user differentiate Official Groups.
- Only one Official Group per organization to let users not confuse among different groups.
- Let all the members get a virtual professional environment of their organizations with the help of Verification Tags and Profile Badges.
- Notification Management for each type of Tabs and Communications for each Group that helps users to prioritize the important ones.
- Exclusive and Flexible Roles and Powers to users depending upon their Roles in their organization.
- Profile Badges.
- In-App Browser for accessing Kiosk, if available, of the Organization.

4.4. ASSUMPTION AND DEPENDENCIES

- There should be internet connection.
- User should have a personal email.
- Domains claiming verification and official status should have an official website and contact details.

4. IMPLEMENTATION

Upon installing the App and opening it the very first time, Login screen pops up that gives the user dual options of signing in as well as registering. As OAuth 2.0 and Google API is integrated into the app, user can use Google Sign In as an alternative to sign in to the dashboard. Hitting the register button, user can register himself/herself on the database. The app checks for email and username validation as well as the password strength that is needed to be at least 8 characters to prevent brute-force attacks. The data is then send to app server to check for the existing usernames and email ids. If there is, user prompted with error and is requested to enter details again. If registration is successful on the app server, the email and password are transferred to the Firebase Authentication Server that encrypts the password with UID and a JSON token. This way, user's credentials are saved nowhere. Firebase Auth helps user to reset password and admin to disable the account.

The domain username is also checked with the app server. If there's no existing domain, user is prompted to create one. User can claim themselves as the admin and add roles to that particular domain. If claimed as admin, verification is requested by the app team from the concerned authority. The designation provided by user in the previous screen is automatically saved to the database.

Upon registering, user is automatically navigated to his/her dashboard using the JSON token generated by Firebase Auth. After further app starts, user is automatically signed in with the help of JSON tokens that are generated during Token Refresh State. These tokens can be used to identify different user segments and to notify them. They are more safe than traditional cookies and related methods. On the Profile Dashboard, users are provided with his profile photo and details as well the functionalities he/she can access.

Verification Badges are displayed beside profile photo and domain name in case they are verified. Clicking on them will display toast with short description about the verification status. User can directly navigate to the group chat room from the dashboard hitting the button that makes automatically scrolls to the latest message.

Group Chat includes all the unblocked members of the domain to send and receive messages. Designation of the member is also displayed beside the sender's name. Enter key is used to insert newlines while messages are sent with the send key. Registered emojis can also sent with supported keyboards. New messages are also received via FCM to the notification bar with a notification sound. From the dashboard, user can also navigate to the Group View from where he/she can view all the members of the same domain. Upon clicking any member, user will be navigated to the member profile. Verification Badges are implemented here as well. Member View displays the same details for the member as the dashboard. User can message and chat with the member using 'Send Message' button. Messages and Chat are implemented with Firebase Real-time Database and Synchronization that syncs messages in real-time.

Connections with the App Server are initiated with Android Volley (introduced in Google I/O 2013) rather than traditional HTTP ASync Tasks. According to the user and member relative hierarchy, user is provided privileges to remove the user from the group and ban him/her from the chat room. If member is removed by the user, next time when the removed member signs in into the app, app prompts an alert and signs him/her out. If blocked from the chat room, member is displayed an alert box when trying to access the chat room. The user is displayed the options of unblocking and add the member again from the same interface. From the Dashboard, user can then sign out from the app if desired. User is navigated to the login screen again after this.

All JSON, Runtime and Connection exceptions are thrown and displayed as a toast in case of failure. Material guidelines and themes are followed for keeping the app updated with the latest market trends. Action bar is initially disabled for all the activities and new App Bar with Dotted Menu is implemented according to the Material Elements mentioned by Google. AppCompact Material EditView fields and buttons are implemented.

5. TESTING REPORT

Developing an android application requires extensive and carefully requirement testing. To get detailed information about the various errors the following tests were carried out and detailed descriptions are given as follows.

Test Environment

The testing was done on Samsung Galaxy S7 Edge running on Android 6.0 Marshmallow.

5.1 UNIT TESTING

TYPE OF TESTING	WILL TEST BE PERFORMED?	COMMENTS/EXPLAINATION	SOFTWARE COMPONENTS
Requirement	Yes	This testing is required because we need to verify Requirements Whether our selected Complete attributes are able to solve the current problem or not.	Complete
Unit	Yes	This testing allows us to whether individuals attributes affects the Application performance or not	Individual Attributes
Security	Yes	This test is performed to check whether the user sensitive data Such as password is secure in the database	Effectively secure

5.2 INTEGRATION TESTING:

Target of this testing is to check whether all modules work together cohesively as per the requirements of the application.

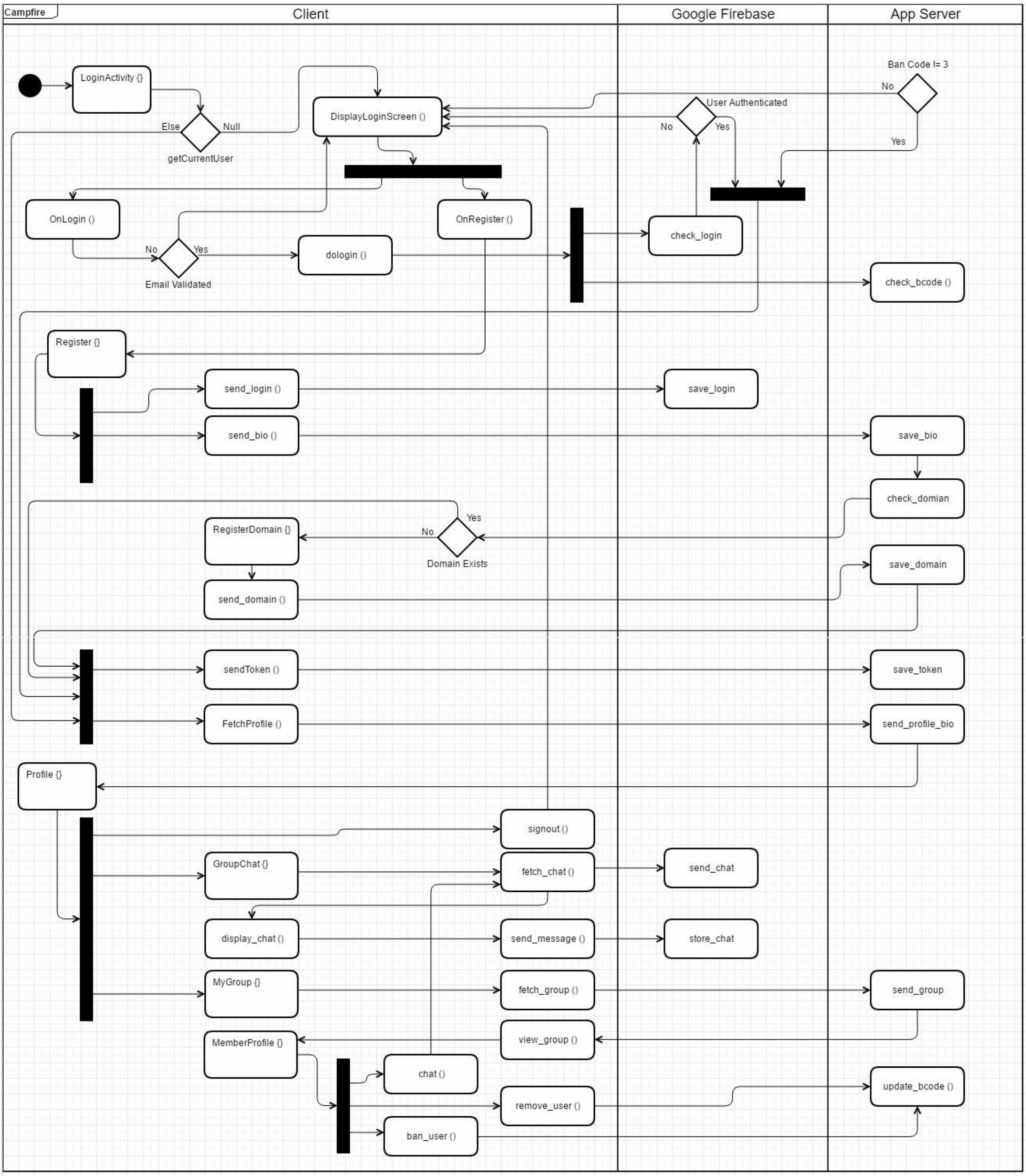
Test Case ID	Input	Expected Output	Status
1.	Application Installation	Installed	Success
2.	Sign Up	Sign Up page load	Success
3.	Fill valid details in sign up form	Form Validations	Success
4.	Fill invalid details at signup	Validation Error	Success
5.	Login with valid combination	Dashboard load	Success
6.	Login with invalid combination	Authentication Error	Success
7.	Sign Up with unregistered organization	Organization not exist	Success
8.	Sign Up in the registered organization Domain.	Registration Successfully	Success
9.	Sending Chat Messages	Message received	Success
10.	New Organization Registration	Create new Domain	Success
11.	Chat Messages in Background	Notification Build	Success
12.	User Registration details	Store in Database	Success
13.	Sign Up via Google Sign-in (OAuth)	Profile Dashboard	Success
14.	Operate with no Internet	Exception Thrown	Success
15.	Load Chat messages	Scroll to bottom	Success
16.	Remove user from Group	Sign out	Success
17.	Ban User from Chat Room	Alert Box	Success
18.	Button change on 17,18	Button Change	Success
19.	Sign out	Login Screen	Success
20.	Verify User/Domain	Verification Badge	Success

5.3 PERFORMANCE TESTING

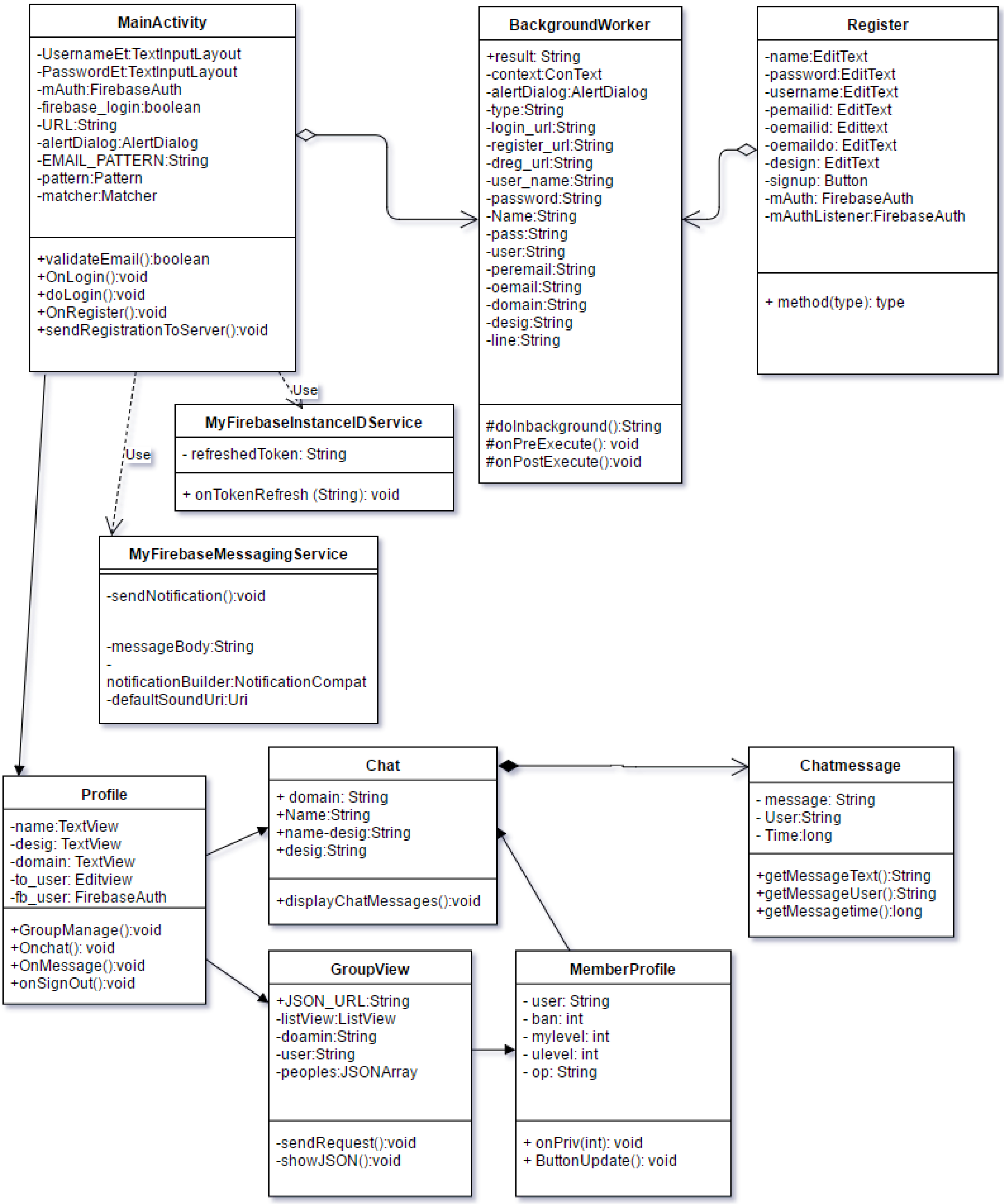
TEST ID	TEST	OUTPUT
1.	Server Operations	Progress bar shown for few seconds
2.	How much response time after a message is sent?	1.5 seconds
3.	How long chat can it save?	1 GB total
4.	How soon/late do you receive the message?	2 seconds on average
5.	Sending lot of messages back to back?	Handled
6.	Min speed required for logging in	2G Or 256 kbps sufficient
7.	Chat box initialization	2-4 seconds
8.	Running on Resolution Devices	No lag
9.	Running on supported but previous versions	Running
10.	Running on unsupported versions	XML Parse errors

4. UML DIAGRAMS

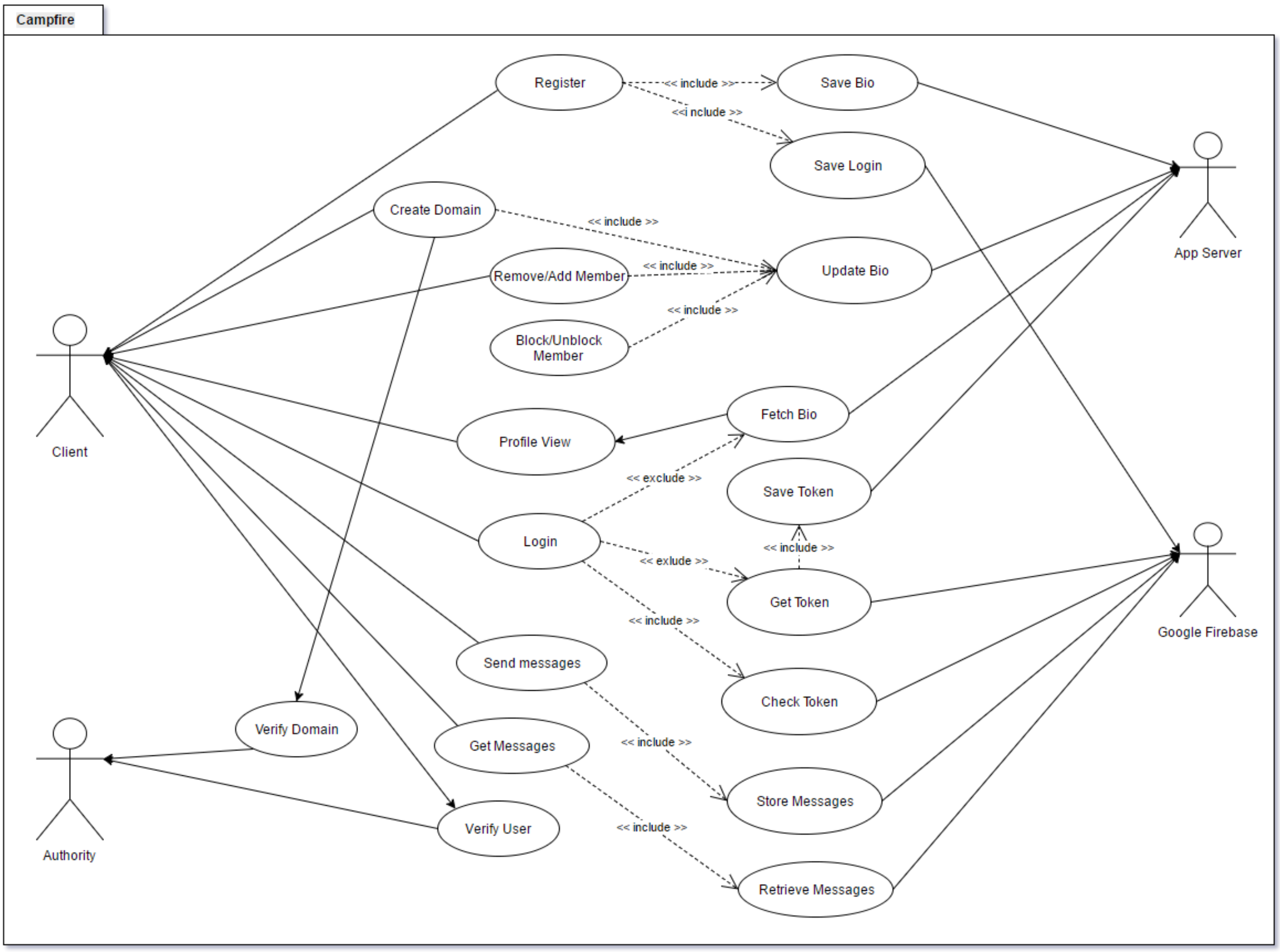
4.1 Activity Diagram



4.2 Class Diagram



4.3 Use-Case Diagram



5. CONCLUSION & FUTURE SCOPE

We have tried our best to integrate all the features and functionalities into the app especially those which we claim makes our app different from others. With no prior knowledge on Android Studio, we started from the scratch 4 months back and have, at last, built the application we desired and what this document mentions about. However, we are aware of the market competition and agree that our app still needs few more vigorous testing and development cycles before deploying it into the market. The technologies are getting updated and vast day by day and we know that our application should have the latest features and elements integrated. Security, Verification and Access Management of the users are the most important aspects for our idea. Although, we have already used UIDs to make user credentials private, end-to-end encryption should be integrated even into the chat module just like the popular IM apps. As far as we have researched, we can claim that the main idea and concept of the app is really new and unique for the market if deployed and integrated as desired. Once the app gets some more time and finance, we believe, we can build a new first-of-its-kind Hierarchy-Based Group Connection and Social Networking Application that virtually implements real-life roles into its environment. For the basic hierarchy, our app server database contains the user privileges, verification status and hierarchy level but for more advanced and state-of-the-art Identity and Access Management technologies like Artificial Intelligence can be integrated which obviously needs advance development of the project. Under Digital India, Government of Idea has just made Open Data available and accessible online. This with AI and pre-data-sets can be used to make verification of a user more reliable. Access Management would need a separate department for managing IAM and associated privileges.

Our application can be used any accessed by any internet user. However, we had started with the idea while considering our nation and the Digital India initiative. Many schools and small-level institutes and organizations can get really benefited from getting themselves online. Even many medium-level organizations too don't have online portals to connect with their members. School and college students and staff members can get connected, have discussions, publish and view important announcements. Verified Groups can help users get rid of the confusion they have while joining a group. Access Management and privileges are based on the real-life hierarchy levels that will grant special users to ban, control and manage their groups. Intranet portals are common in high-level organizations which helps users to connect and posts stuff related to the organization. Campfire can help them do the same while sitting at their homes outside their office network. Basic and even most popular IM apps lack isolation of tabs based on one's requirements. One might not want to use a messaging services for social networking but for only important announcements. Notification Management Control Settings can help users define different notifications like ringtone, alerts, led and custom functions to each type of chat room and notification services. This way Campfire is rightly suitable for both personal and professional use. Groups already made can be searched and refined based on their verification status that can help users decide among different groups made for the same organization or domain as well as know and join the official one. Campfire only allows one verified and official group for a particular real-life domain, for example, JIIT can have only one official group on our app that has been already verified by the JIIT managing head or equivalent. This is presently done manually to avoid any coding and machine exploits but in future can be extended to Artificial Intelligence and Deep Learning. However, multiple groups can be made based on interest and requirements, for example, JIIT CSE Staff Group or JIIT Gaming Hub.

Hence, we believe, this project surely has a future scope and we will continue as a team for deploying this into the market as soon as possible.

6. REFERENCES

1. Android Volley
   - Google I/O 2013 Presentation [PDF]
   - Google I/O 2013 Video
   - Volley | Google Developers

2. Firebase by Google
   - Authentication
   - Real-time Database
   - Cloud Messaging

3. Material Guidelines by Google
   - Creating Apps – Getting Started
   - Elevation and Shadows
   - Using the Material Theme
   - Colour Palette

4. Research Papers & Related Research for IAM
   - Security and access management system for web-enabled and non-web-enabled applications and content on a computer network – Google Patent US 6460141 B1

- Havard University – IAM Program Plan [PDF]