Assessment 3

Sprint 2: Project Handover and Presentation

SIT708 – Mobile Systems Development

Name: Viseth Kong

ID: 218416405

Executive Summary

In the present, COVID-imposed lockdowns have an impact on millions of people all over the world. Face-to-face activities and classes are difficult for people to participate in because of physical distance. Mental health problems like as sadness and anxiety are on the rise because of the increased isolation that comes with physical distance, which helps keep individuals safe from the coronavirus. As a result of the COVID lockdowns, I was motivated to create an app that would let people learn new skills during their leisure time. This lockdown period has allowed me to acquire a lot of new skills related to software development using online platforms like SkillShare and Udemy. However, I've had to pay for each course I've taken, and it may be tough to get in touch with the experts.

The project concept focuses on building a community-driven online platform for sharing skills. It brings people together with the goal of sharing information, offering assistance, and reducing feelings of isolation. Users may find each other by searching for activities they like and discovering others who share their interests.

SocialCollab

SocialCollab is a mobile android application that enables guided, productive interaction between users. It focuses on two types of users: those who want to offer their knowledge ("tutors") and others who wish to acquire knowledge ("students"). Whether they are tutors, students, or both, users will find everything necessary to advertise, discover, engage, and communicate. Users may join up and establish a profile detailing the skills they desire to share or teach. Their profile might have a video introduction, a portfolio, and other elements. When prospective students find a tutor they like, they may request to communicate with them by text, audio, or video call, or they can schedule a session. This enables users to interact in real time for customized one-on-one sessions.

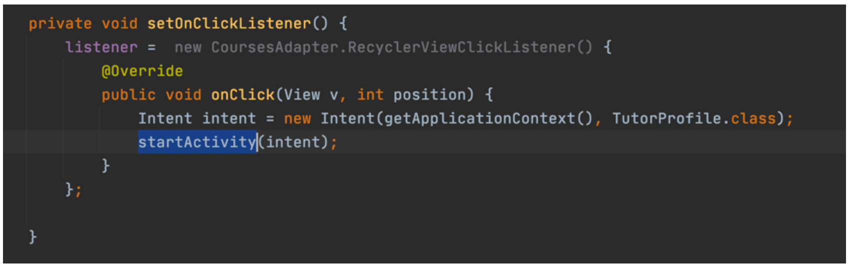
Development Process

As a starting point for the development of SocialCollab, I identified use cases and user stories that were not seen in other applications. The UI/UX design phase of SocialCollab began when the use cases and core functionality were finalized. Wireframes for SocialCollab were created using a web designing program called Figma. In order to meet the requirements of the use cases, it was determined that SocialCollab needed at least 24 different pages for users to navigate. The front-end development of SocialCollab's UI began using Android Studio. Afterward, Java programming language was used to implement SocialCollab behaviors and functions. The back-end development process was the next step, and it included connecting the project to a firebase database so that it could perform such a variety of functions.

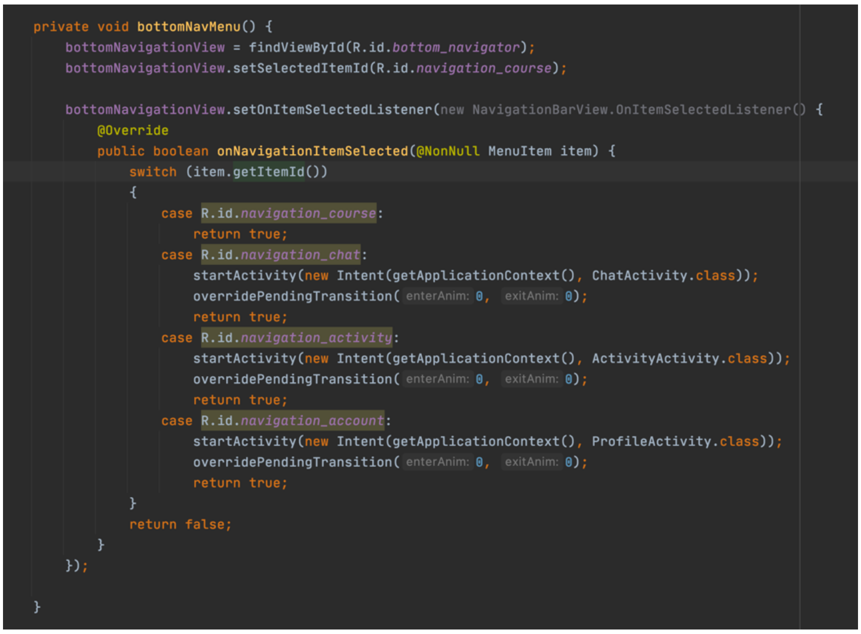
Primary Features & Coding Concepts

SocialCollab's Navigation

Activities were used within SocialCollab to support in managing and displaying the various pages. Within each activity, the startActivity() method is used to declare navigation between activities. Furthermore, to assist in declaring each bottom navigation menu behavior, an Menu XML file containing each navigation action was created. When a user chooses to change between sections, the main activity calls the BottomNavMenu method and its associated navigation action.



```
private void setOnClickListener() {
    listener =  new CoursesAdapter.RecyclerViewClickListener() {
        @Override
        public void onClick(View v, int position) {
            Intent intent = new Intent(getApplicationContext(), TutorProfile.class);
            startActivity(intent);
        }
    };

}
```

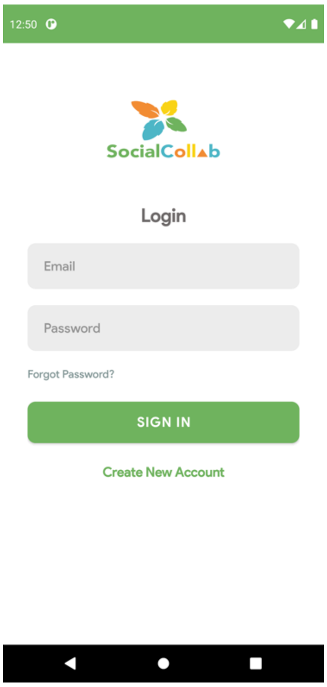


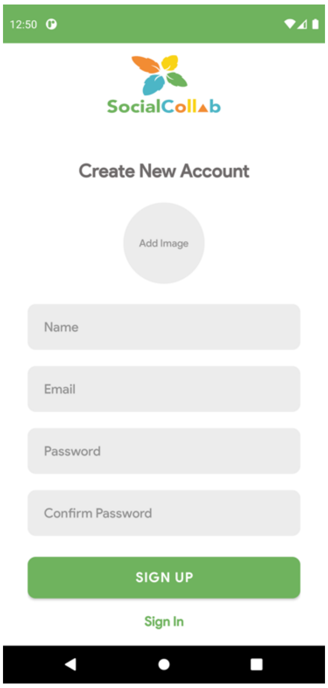
```
private void bottomNavMenu() {
    bottomNavigationView = findViewById(R.id.bottom_navigator);
    bottomNavigationView.setSelectedItemId(R.id.navigation_course);

    bottomNavigationView.setOnItemSelectedListener(new NavigationBarView.OnItemSelectedListener() {
        @Override
        public boolean onNavigationItemSelected(@NonNull MenuItem item) {
            switch (item.getItemId())
            {
                case R.id.navigation_course:
                    return true;
                case R.id.navigation_chat:
                    startActivity(new Intent(getApplicationContext(), ChatActivity.class));
                    overridePendingTransition( enterAnim: 0,  exitAnim: 0);
                    return true;
                case R.id.navigation_activity:
                    startActivity(new Intent(getApplicationContext(), ActivityActivity.class));
                    overridePendingTransition( enterAnim: 0,  exitAnim: 0);
                    return true;
                case R.id.navigation_account:
                    startActivity(new Intent(getApplicationContext(), ProfileActivity.class));
                    overridePendingTransition( enterAnim: 0,  exitAnim: 0);
                    return true;
            }
            return false;
        }
    });

}
```

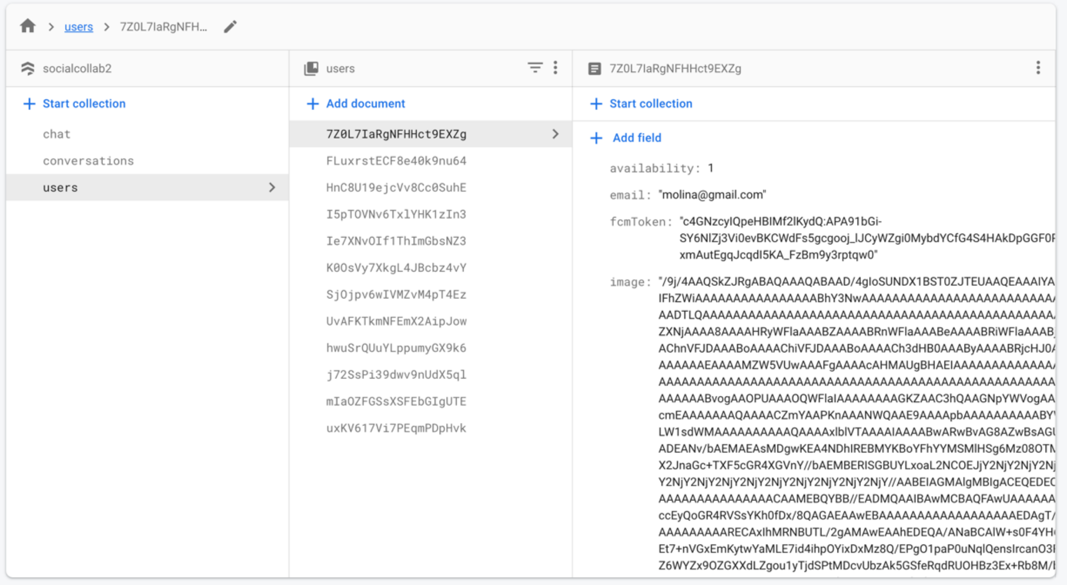
Login or Register Account - Firestore Database

One of SocialCollab's use cases was to allow users to create an account and log in to an existing account using their login and password credentials. Firebase was integrated into SocialCollab since it gives Android Studio with an efficient connection to a project and its databases. Firestore database was implemented for the login and registration features. As soon as they open SocialCollab, users are asked to sign in using an email address or password. The user can sign up if the user don't have an account by clicking on the Sign Up button, which takes the user to the registration page.

Users' email addresses, passwords, and unique IDs are stored in the Firestore database after they've registered an account.







chat
conversations
users

7Z0L7IaRgNFHHct9EXZg
FLuxrstECF8e40k9nu64
HnC8U19ejcVv8Cc0SuhE
I5pTOVNv6TxlYHK1zIn3
Ie7XNvOIf1ThImGbsNZ3
K0OsVy7XkgL4JBcbz4vY
SjOjpv6wIVMZvM4pT4Ez
UvAFKTkmNFEmX2AipJow
hwuSrQUuYLppumyGX9k6
j72SsPi39dwv9nUdX5ql
mIaOZFGSsXSFEbGIgUTE
uxKV617Vi7PEqmPDpHvk

availability: 1

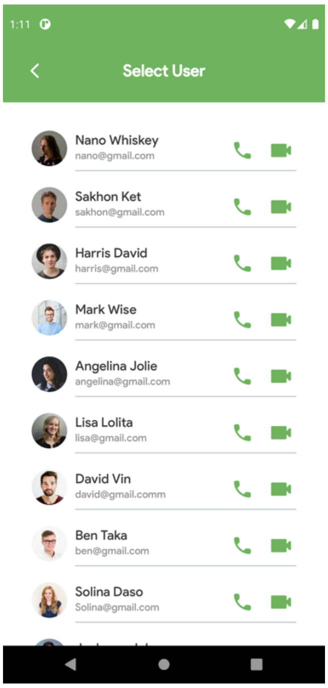
email: "molina@gmail.com"

fcmToken: "c4GNzcyIQpeHBIMf2lKydQ:APA91bGi-SY6NlZj3Vi0evBKCWdFs5gcgooj_lJCyWZgi0MybdYCfG4S4HAkDpGGF0F
xmAutEgqJcqdI5KA_FzBm9y3rptqw0"

image:
"/9j/4AAQSkZJRgABAQAAAQABAAD/4gIoSUNDX1BST0ZJTEUAAQEAAAIYA
IFhZWiAAAAAAAAAAAAAAAABhY3NwAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AADTLQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
ZXNjAAAA8AAAAHRyWFlaAAABZAAAABRnWFlaAAABeAAAABRiWFlaAAAB
AChnVFJDAAABoAAAAChiVFJDAAABoAAAACh3dHB0AAAByAAAABRjcHJ0A
AAAAAAEAAAAMZW5VUwAAAFgAAAAcAHMAUgBHAEIAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAABvogAAOPUAAAOQWFlaIAAAAAAAAGKZAAC3hQAAGNpYWVogAA
cmEAAAAAAAQAAAACZmYAAPKnAAANWQAAE9AAAApbAAAAAAAAAAABY
LW1sdWMAAAAAAAAAAAQAAAAxlblVTAAAAIAAAABwARwBvAG8AZwBsAGU
ADEANv/bAEMAEAsMDgwKEA4NDhIREBMYKBoYFhYYMSMlHSg6Mz08OTM
X2JnaGc+TXF5cGR4XGVnY//bAEMBERISGBUYLxoaL2NCOEJjY2NjY2NjY2Nj
Y2NjY2NjY2NjY2NjY2NjY2NjY2NjY2NjY2NjY//AABEIAGMAlgMBIgACEQEDEC
AAAAAAAAAAAAAACAAMEBQYBB//EADMQAAIBAwMCBAQFAwUAAAAAA
ccEyQoGR4RVSsYKh0fDx/8QAGAEAAwEBAAAAAAAAAAAAAAAAAAEDAgT/
AAAAAAAAARECAxIhMRNBUTL/2gAMAwEAAhEDEQA/ANaBCAIW+s0F4YH
Et7+nVGxEmKytwYaMLE7id4ihpOYixDxMz8Q/EPgO1paP0uNqlQensIrcanO3I
Z6WYZx9OZGXXdLZgou1yTjdSPtMDcvUbzAk5GSfeRqdRUOHBz3Ex+Rb8M/

Display and Select User to interact with from the User List – Recycler View

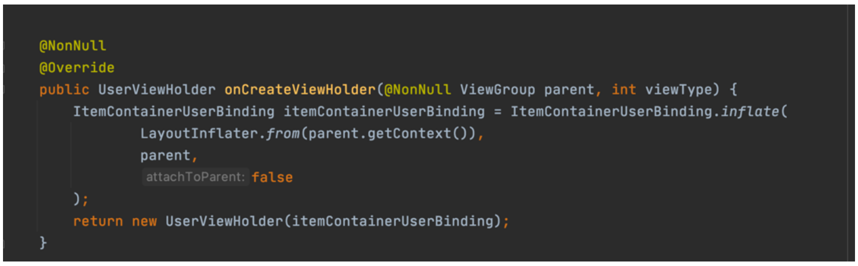
This feature show a user list in recycler view by retrieving it from the Firestore Database which allows user to select and interact another user. I created a User Adapter class to view the user data retrieved from Firestore Database.



Select User

Nano Whiskey	nano@gmail.com
Sakhon Ket	sakhon@gmail.com
Harris David	harris@gmail.com
Mark Wise	mark@gmail.com
Angelina Jolie	angelina@gmail.com
Lisa Lolita	lisa@gmail.com
David Vin	david@gmail.comm
Ben Taka	ben@gmail.com
Solina Daso	Solina@gmail.com

I used viewBinding for this adapter class, the binding class for each XML layout will be generated automatically. And ‘ItemContainerUserBinding’ class is automatically generated from the layout file called ‘item_container_user’.

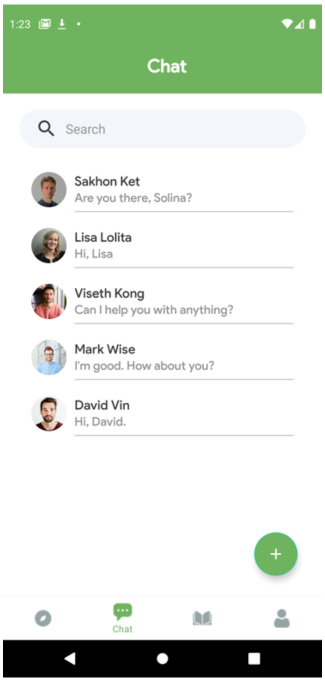


```
@NonNull
@Override
public UserViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
    ItemContainerUserBinding itemContainerUserBinding = ItemContainerUserBinding.inflate(
            LayoutInflater.from(parent.getContext()),
            parent,
             attachToParent: false
    );
    return new UserViewHolder(itemContainerUserBinding);
}
```

Chat System Feature – Firestore Database

Chat system allow users to communicate with each other with real time messages. First, I designed layouts related to chat screen like chat screen, sent message layout, received message layout, etc. After that, I prepared data for real-time chat like chat message model, chat adapter (RecyclerView's adapter), etc. Then, I implement a real-time chat facility with Firestore Database that will cover various features such as

- Send a message and receive a messages
- Show the last message in recent conversations
- Update recent conversations real time
- Conversation will move to the top when a new message arrives
- Handle user availability like the user is online or not
- Push notifications when receiver is not online/available.



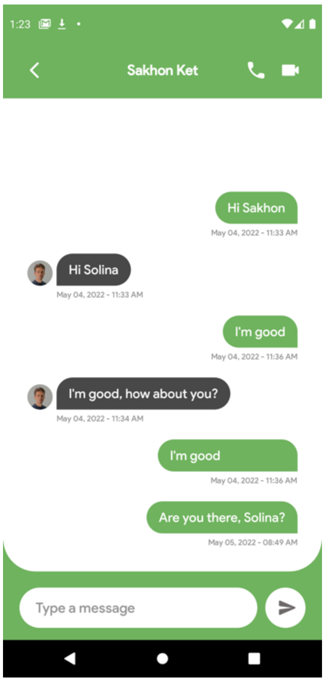
Sakhon Ket
Are you there, Solina?

Lisa Lolita
Hi, Lisa

Viseth Kong
Can I help you with anything?

Mark Wise
I'm good. How about you?

David Vin
Hi, David.



Hi Sakhon
May 04, 2022 - 11:33 AM

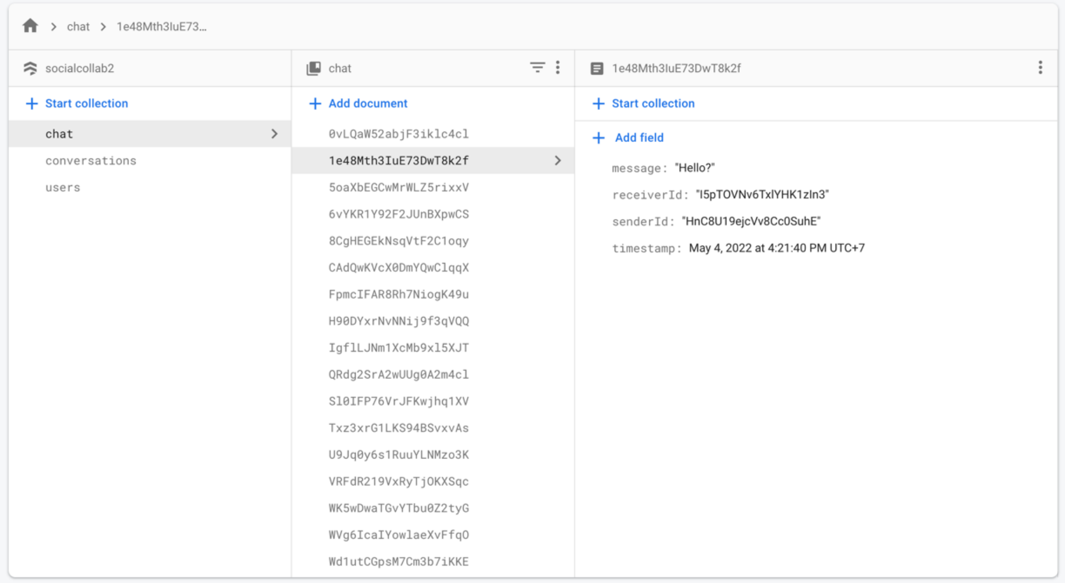
Hi Solina
May 04, 2022 - 11:33 AM

I'm good
May 04, 2022 - 11:36 AM

I'm good, how about you?
May 04, 2022 - 11:34 AM

I'm good
May 04, 2022 - 11:36 AM

Are you there, Solina?
May 05, 2022 - 08:49 AM



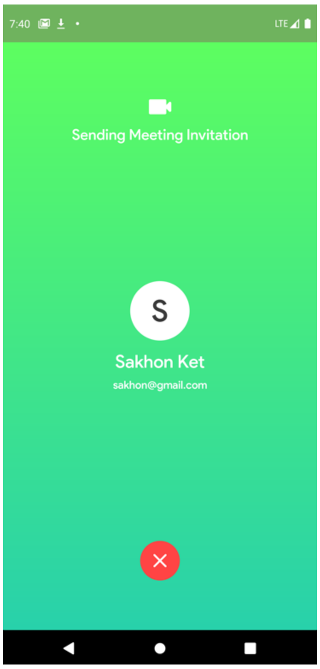
```
chat
conversations
users
```

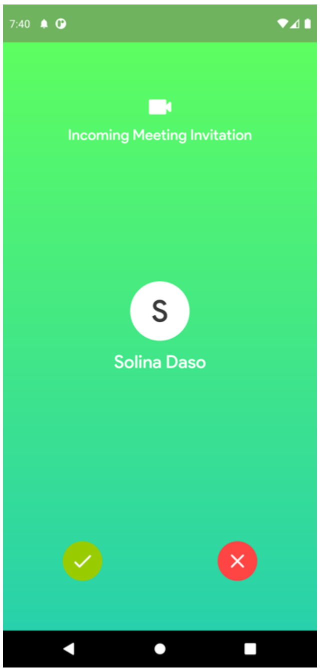
```
0vLQaW52abjF3iklc4cl
1e48Mth3IuE73DwT8k2f
5oaXbEGCwMrWLZ5rixxV
6vYKR1Y92F2JUnBXpwCS
8CgHEGEkNsqVtF2C1oqy
CAdQwKVcX0DmYQwClqqX
FpmcIFAR8Rh7NiogK49u
H90DYxrNvNNij9f3qVQQ
IgflLJNm1XcMb9x15XJT
QRdg2SrA2wUUg0A2m4cl
Sl0IFP76VrJFKwjhq1XV
Txz3xrG1LKS94BSvxvAs
U9Jq0y6s1RuuYLNMzo3K
VRFdR219VxRyTjOKXSqc
WK5wDwaTGvYTbu0Z2tyG
WVg6IcaIYowlaeXvFfqO
Wd1utCGpsM7Cm3b7iKKE
```

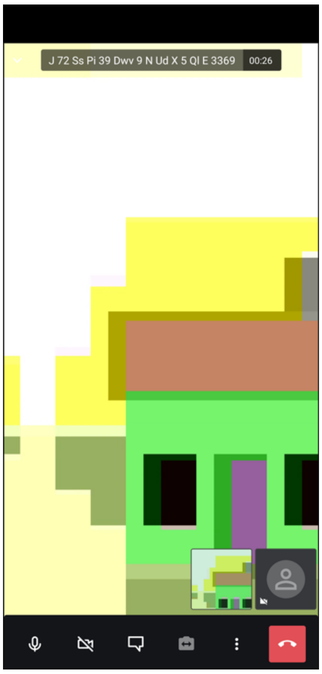
message: "Hello?"
receiverId: "I5pTOVNv6TxlYHK1zIn3"
senderId: "HnC8U19ejcVv8Cc0SuhE"
timestamp: May 4, 2022 at 4:21:40 PM UTC+7

Audio and Video conference - Firebase Authentication Database

This feature allows users in different locations to hold face-to-face meetings without having to move to a single location together. First, I set up Retrofit which is a networking library and it can help in doing some additional configurations. Then, I initiated a video meeting by sending a remote message using firebase cloud messaging from an android device, and used Retrofit for API calls. implement the facility to accept, reject & cancel the meeting invitation using firebase cloud messaging. I also implemented the facility to accept, reject & cancel the meeting invitation using firebase cloud messaging. For video meeting and audio meeting, I used Jitsi Meet Android SDK which is a fully encrypted, 100% open source video conferencing solution.







Challenges During Development

- When I started working on SocialCollab, I had to determine whether it would be built as a single activity with corresponding child fragments, or as a collection of separate activities that interact with each other. At first, I decided to use a large number of fragments that would inherit the main activity's background image and layout. It was challenging to read and write to Firestore database after implementing Firebase and other external libraries. Furthermore, the vast majority of Firebase guides and tutorials explained and demonstrated how to import Firebase using activities rather than fragments. As a result, towards the end of the SocialCollab development process, I changed most fragments to activities in order to enable the use of Firebase and other libraries.
- The connection between SocialCollab and Firebase's databases was an issue that had to be addressed. In order to log in, register a user, and store and retrieve records like chat messages, user availability, and FCM tokens, SocialCollab relies on Firebase to authenticate the user. One of the challenges I encountered was ensuring that data like as the user's profile picture, username, and email address should be recorded in the Firebase storage when they logged in. Consequently, while two users are communicating, the data should be read and written at the same Firebase node. After connecting the SocialCollab app to Google Firebase and doing tests that include signing up, logging in, or sending text messages, the data should be reflected in the database.
- Creating dynamic displays for recyclers was an additional challenge for me. SocialCollab's recycler views are essential for efficiently scrolling over a sorted list of data. In order to display a list of users and chat messages in a recycler view layout, the data has to be parsed and linked to the Firestore Database.
- Another difficulty I faced was developing video conferencing on the Jitsi Meet platform, which required extensive research and analysis of the platform's system

architecture and infrastructure as well as the video conference application itself. So, meeting invitations are sent to each other using an API Service of the Retrofit library, which uses the FCM token that is defined when a user logs into the system. This service listens to the database for FCM tokens to determine whether users are currently online. When a person is online, an invitation to a meeting is sent.

https://www.researchgate.net/publication/355163234_Development_of_Android_Based_Mobile_Video_Conference_Application_Using_Jitsi_Meet_Platform

Future Development

- The UI and UX design will be improved further to enhance the user experience by adding some of the modifications to each page and feature, with the intention of reducing the amount of screen time that is not essential.
- Add all available options to the Settings page, including ‘Edit profile’, ‘Put account in private mode’, ‘Dark mode’, and add a variety of other options.
- It is planned to implement a search tool that will enable users to seek for tutors or courses.
- Optimizing Firebase data retrieval by cleaning and enhancing database queries will reduce the amount of time it takes for SocialCollab and Firebase to interact and process each other.
- The recommendation course feature will be deployed to make it easier for users to locate the courses in which they are interested.
- Implement the potential to start a video meeting (conference) that allows multiple users to attend the conference by using Jitsi Meet.

Conclusion

SocialCollab is a platform that provides users the opportunity to share their knowledge to other users who are wanting to learn new skills or discover new hobbies. Beside this, I hope that SocialCollab can enhance and take care of every user's physical health, as well as enrich people's communication and activities, so that they can retain a healthy physique and an optimistic attitude to life.

GitHub Link to SocialCollab Repository

https://github.com/visethkong/SIT708---Mobile-Systems-Development

YouTube Link to SocialCollab Presentation & Demonstration

https://youtu.be/cCP6zpU9dV0