Capstone_Stage1 (Let's chat app)

- Description
- Intended User
- Features
- User Interface Mocks
  - Screen 1
  - Screen 2
- Key Considerations
    - How will your app handle data persistence?
    - Describe any corner cases in the UX.
    - Describe any libraries you'll be using and share your reasoning for including them.
    - Describe how you will implement Google Play Services.
- Next Steps: Required Tasks
  - Task 1: Project Setup
  - Task 2: Implement UI for Each Activity and Fragment
  - Task 3: Your Next Task
  - Task 4: Your Next Task
  - Task 5: Your Next Task

GitHub Username: amkraneYoussef

Let's Chat app

Description

Welcome to this app. By joining Let's Chat app community to chat with local people. It is a free app to make new friends as well as connecting with other people from nearby cities. The goal is to make it easy for people to make new connections

Technical description

- This project will be developed using only JAVA programming language
- **Admob** will be used to serve ads to the user.
- All **resources** will be stored in the res folders at the appropriate file for each resource (strings.xml for strings colors.xml for colors styles.xml for app theme etc..)
- **IntentService** will be used to send data to widget.
- Minimum SDK will be 19, **RTL** is supported and will be enabled in all layouts and '**autoMirrored'** will be set for drawables that require to be mirrored

Libraries

This app will use only stable releases off all libraries:

- `glide:4.9.0`
- `circleimageview:3.0.0`
- `design:28.0.0`
- `firebase-database:16.0.6`
- `firebase-auth:16.1.0`
- `play-services-auth:16.0.1`
- `firebase-core:16.0.7`
- `play-services-ads:17.2.0`

Intended User

All local people interested in making new friends.

Features

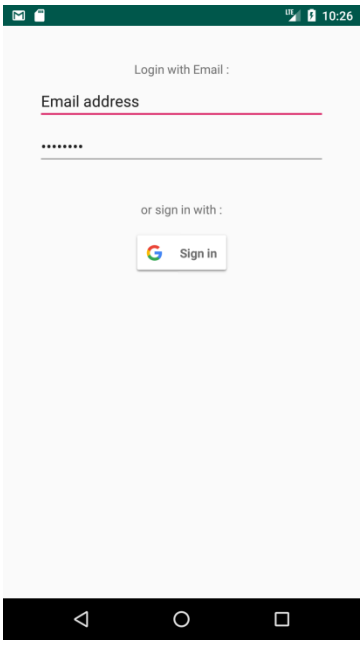
- Create and login to an account
- Add profile information (Photo, Full Name, username, birthdate, gender, current city)
- Get available users
- Send and receive messages
- Store messages in the phone

Accessibility

- All texts should size should be defined in SP.
- ContentDescription attribute will be used to the describe the content of imageViews (except decorative images)

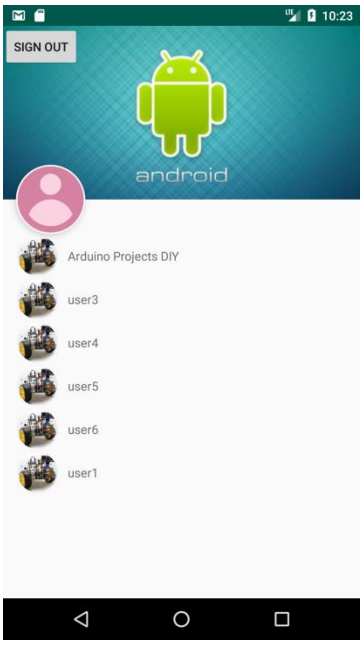
User Interface Mocks

Screen 1 : Login screen



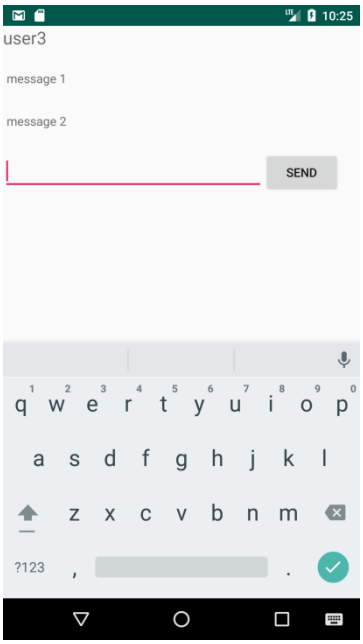
This screen allows the user to register using an email and password or sign in with Google;

Screen 2 : Online users list

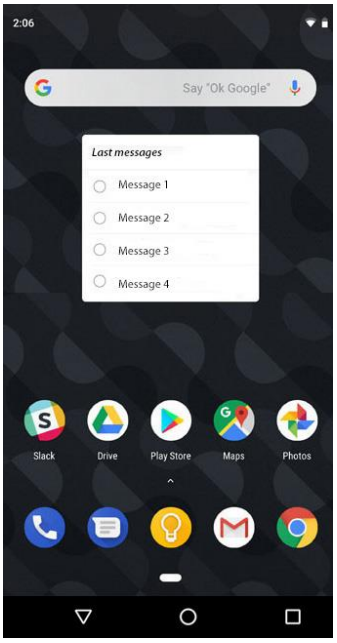


This screen lists online people and allows the user to chat with anyone.

Screen 3 : Activity where the user sends and receives messages



Screen 2 : Widget for last messages



Key Considerations

How will your app handle data persistence?

This application relies on the use of Firebase Realtime Database to store messages and users data.

Describe any edge or corner cases in the UX.

- Login activity should be loaded when the user is not logged in.
- App user profile should be created if it does not exist in the database
- When the user is chatting with another one, he must be notified of new messages send by other users and notifications should be deleted when the user reads the messages.
- The app should only show last messages and should allow the user to scroll up to view older messages using continuous scrolling.

Describe any libraries you'll be using and share your reasoning for including them.

Material design library
Glide to manage images.
Circle image view library (https://github.com/hdodenhof/CircleImageView)

Describe how you will implement Google Play Services or other external services.

Google play services will be used to authenticate users

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

Create an android studio project
- Add activities to the project
- Configure dependencies to the required libraries

Create Firebase account.
- Add an android project
- Link the project with my chat app

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
- Build UI for login activity
- Build UI for **Messaging** Acitivity

Task 3: Code login activity

Give the user a possibility to :

- Create an account or use google account
- Sign in to an account

Task 4: Code main activity

In main activity:

- The use can view online users
- Access his chat list
- Get notified if there are any new messages

Task 5: Coding messaging activity

- When the user selects an online user, messaging activity loads where he can start a chat with the remote user

Task 6: Widget implementation

- Define an XML file describing the AppWidgetProviderInfo object
- Extend the class `AppWidgetProvider` to link the app to the widget
- Creating the App Widget and item Layout (list will be loaded to a **GridView**)
- Add widget to manifest
- Create a `RemoteViewsFactory` and a `RemoteViewsService` and register the service
- Start the service from onUpdate of the widgetProvider
- A click on any message of the widget should initiate a chat with the remote user