GitHub Username: https://github.com/huse

Messaging app

Description
Messenger for smart phones, sending text to friends around the world.
App will be written solely in the Java Programming Language.

Intended User

This app is for all smart phones around the world to communicate with each.

Features

List the main features of your app. For example:
- Saves information
- Send and receive messages and picture or video
- Show and save messages in recycle view

User Interface Mocks

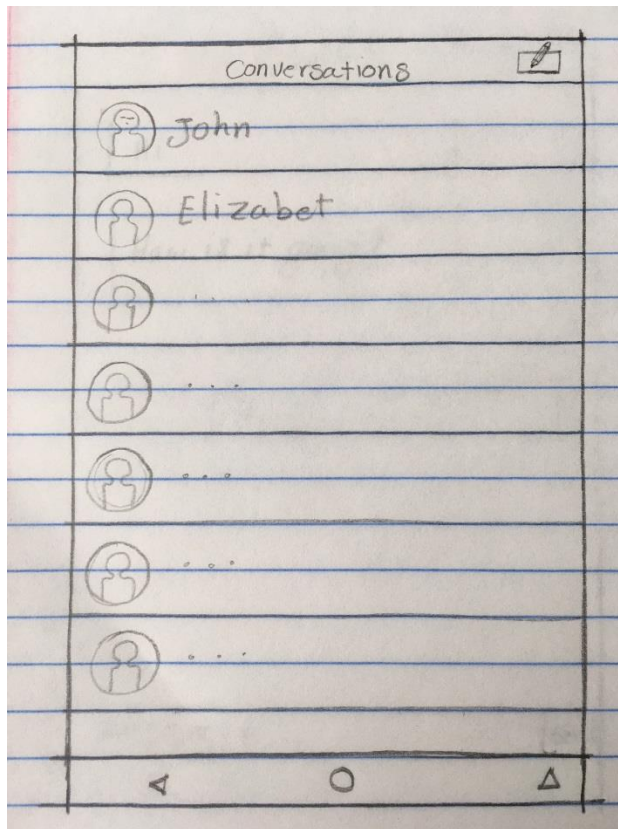These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, www.ninjamock.com, Paper by 53, Photoshop or Balsamiq.
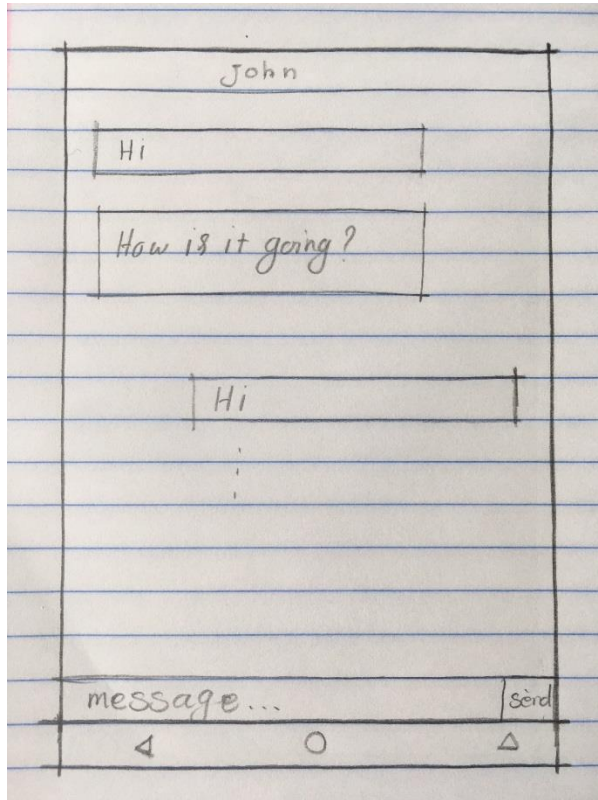
Screen 1
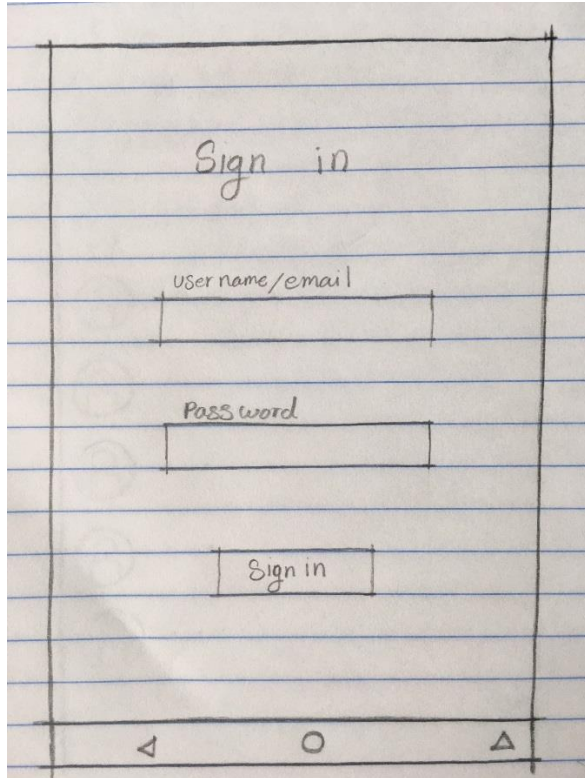
Sign in UI has two textEdit for password and username and one button to sign in.
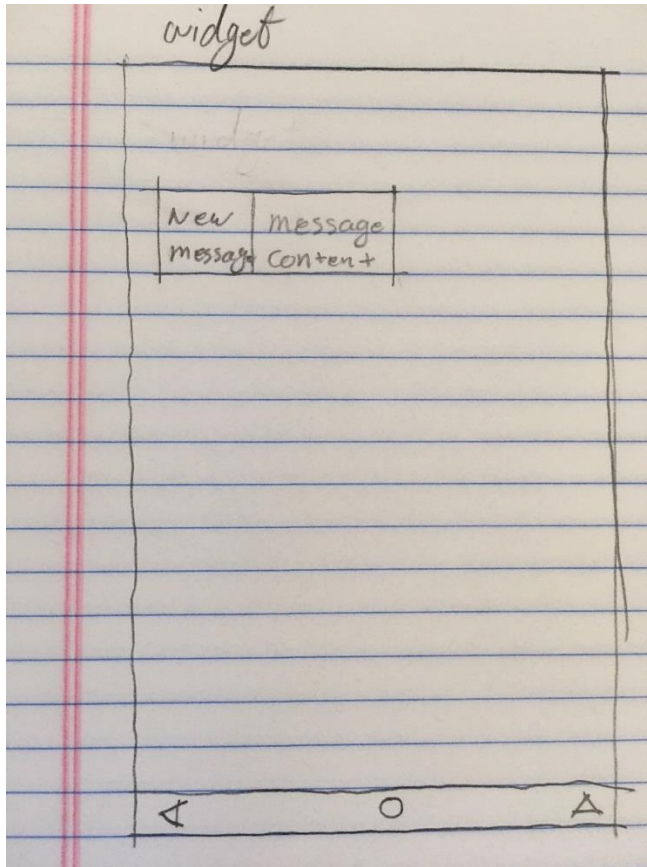
Screen 2

MainActivity UI contains conversation history in Recyclerview and new conversation button at top right corner.

Screen 3

Detail activity fragment UI that user can send and receive messages.

Screen 4

Widget

Key Considerations

Libraries will be used

	Software / libraries to use	Version / or higher
	android studio	3.1.2
	Gradle	4.4
	Firebase	15.0.0
	firebase-auth	15.1.0
	firebase-core	15.0.2
	firebase-messaging	15.0.2
	firebase-client-android	2.5.2
	firebase-jobdispatcher	0.8.5
	recyclerview	v7:27.1.1
	constraint:constraint-layout	1.1.0

How will your app handle data persistence?

For saving and handling data the app uses Firebase Realtime Database.

Describe any edge or corner cases in the UX.

In sign in when sign in pressed the user name and password send to server to authenticate user and return results to app to give access to user.
In main activity messages sync with database.
In detail activity messages retrieve from server.
In detail activity when send pressed the message content will be sent to other user.

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

App uses Picasso to handle the loading and caching of images.
App uses Butterknife to reduce coding.
App uses Android support library to design UI.

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

Firebase database for storing user information and messages.

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

In first step all libraries will be configured in gradle.
- Configure libraries
- Install dependencies and libraries in Gradle.

Task 2: Implement UI for Main activity

- Create layouts for MainActivity,
- Build UI for MainActivity include start new conversation, recyclerview for list of chat history.

Task 3: Implement UI for **Login activity**
- Create layout for login activity
- Build UI for login activity.

Task 4: Implement UI for **Detail activity**

- Create layout for fragment activity
- Build UI fragment to show each conversation

Task 5: **Data base**

- Implementing Firebase Realtime Database
- Creating data base for storing data and information.

Task 6: **Content provider**

- Implement content provider.