IEMS5722 Mobile Network Programming and Distributed Server Architecture (Fall 2024)
Assignment 2
Expected time: 6 hours

Learning outcomes:
1. To learn how to construct the user interface of an Android app 2. To learn how to receive and handle user input 3. To understand page transition in Android 4. To learn how to use LazyColumn and data class

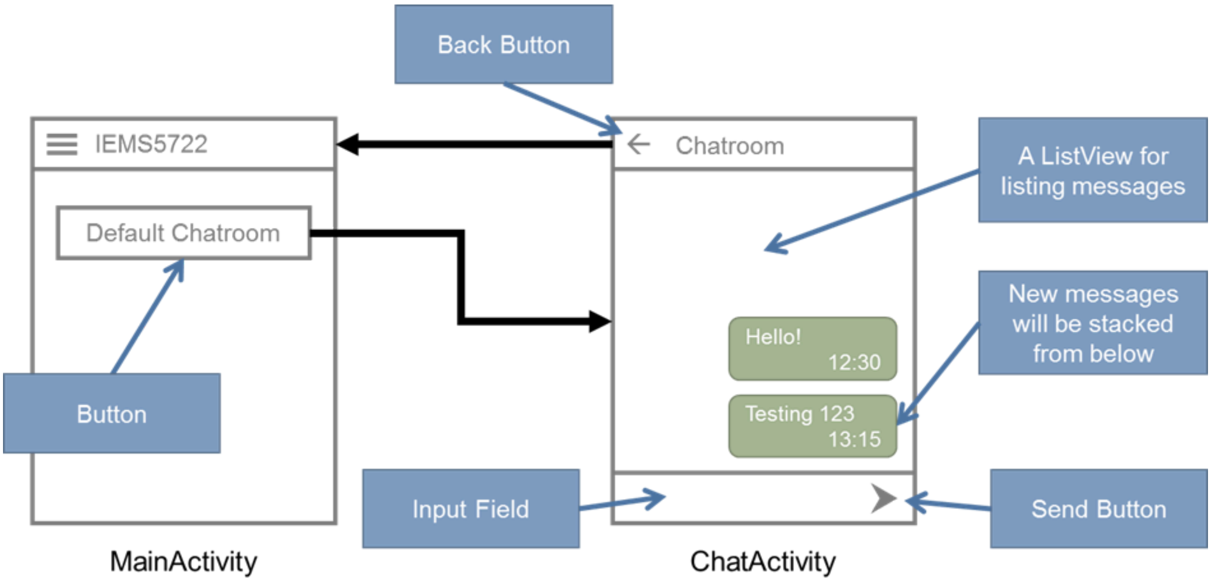
Instructions:

1. Do your own work. You are welcome to discuss the problems with your fellow classmates. Sharing ideas is great, and do write your own explanations/comments.
2. If you use help from the AI tools, e.g. ChatGPT, write clearly how much you obtain help from the AI tools. No marks will be taken away for using any AI tools with a clear declaration.
3. All work should be submitted onto the blackboard before the due date.
4. You are advised to submit a .pdf file and a .zip file containing all your work.
    - 1155xxxxxx_Assignment2.pdf: The short report for your work. We will grade your work based on the short report.
    - 1155xxxxxx_Assignment2.zip: The zip file containing your Android Studio project. In general, we will not check this archive in grading. In case, we found some problems in your report (meaning that you already lost some points in your report), we will refer to your project source code.
    - 1155xxxxxx is your student ID.
5. Do type/write your work neatly. If we find some problems in the screenshots in your report, you will lose some points, even if those contents are in the source files.
6. If you do not put down your name, student ID in your submission, you will receive a 10% mark penalty out of the assignment 2.
7. Late submissions will receive a 30% mark penalty.
8. This assignment accounts for 6% of your final grade.
9. Due date: 10th October, 2024 (Thursday) 23:59.

Instructions:

In this assignment, you are going to develop a “mock-up” instant messaging app using a new “empty activity” project. You should use Jetpack Compose to build the user interface (UI) for both MainActivity (selecting multiple chatrooms with buttons) and ChatActivity (a messenger application). The app should have a chat interface similar to those found in common messaging apps like WeChat, WhatsApp or Facebook Messenger.

The figure below shows a draft of the UI of the app.



The app has two activities. One activity is the MainActivity. The MainActivity has a button called “Default Chatroom”. When the user clicks on the button, the app should redirect the user to another activity called the ChatActivity. The ChatActivity has an interface containing a LazyColumn, which displays the past messages in the chatroom, an input area where the user can input his or her own message, and a “Send” button to allow the user to submit the message.

In this assignment, you do not have to worry about creating a real instant messaging app. This assignment is mainly for you to get familiar with the UI components in Android development.

You can style the UI components in your own way, but the app you submit in this assignment should have a layout the same as what is illustrated in the figure above.

Tasks:

1. The app contains two activities, which are MainActivity and ChatActivity. (10%)
2. There is a button on MainActivity which will allow the user to enter the ChatActivity. (10%)
3. The ChatActivity has a back button on the top, allowing the user to go back to the MainActivity. (10%)
4. The ChatActivity consists of a LazyColumn for listing past messages, an EditText for user input, and an ImageButton or ImageView for the user to send the message. (15%)
5. The ChatActivity displayed some past messages using a hard code function. (10%)
   - These hard code messages should be displayed in the chatroom at the beginning. It means, the messages in the chatroom will always be the same after reloading the app.
   - You will remove this part in the next assignment.
6. In the ChatActivity, only your messages are aligned to the right, while the remaining messages are aligned to the left. (10%)
7. Each message item in the LazyColumn should contain three elements, which are the message itself, the time (HH:MM) when the message is sent and a chat image background behind the text. (15%)
8. When the user clicks on the send button, the app should perform the following actions: (20%)
   - Check if the user has input any text (i.e. check whether the input area is empty)
   - If the user has input some text, add a new item containing the message to the end of the LazyColumn; and then clear the input area

Note that you do not need to store the messages in any database systems. We will see how to make the app more realistic in the next assignment.

If you decide to use Java or other Kotlin approaches to design your chatroom, you will need to adjust the description of the screenshots on your own.

Resources:

1. Layouts
   https://developer.android.com/develop/ui/compose/layouts
2. Controls (Buttons)
   https://developer.android.com/develop/ui/compose/components/button
3. Input Events (States)
   https://developer.android.com/develop/ui/compose/state
4. Intent and Sharing
   https://developer.android.com/develop/ui/compose/glance/user-interaction