Introduction

This last challenge is supposed to culminate your work in this course. In the previous challenge, you learned how to create more advanced interfaces, how to store and get persistent data from Internal Storage and how to connect with Firebase. In this challenge we want you to integrate together a messaging application and an IoT device, to interact directly with your smartphone using MQTT. You will develop a messaging application which will use MQTT to send and receive messages, also notifying the user about new messages - additionally, an Arduino will be used to show incoming messages from specific people.

You should implement this challenge in Wokwi (https://wokwi.com/). For any questions or doubts, you can send an e-mail to your professors.

Milestone 3

You can use the basic code found at https://wokwi.com/projects/414285391320977409 - there are a lot of comments and mostly everything is already there for you to mess around. It contains a red LED, an LCD and part of the connection to MQTT for you to use.

This milestone is to be done in groups and the objectives are:

- Develop an application where you can send and receive text messages. This application should contain at least four fragments (dynamic). If you use only activities or static fragments, you will be penalized by -5%.
    - One fragment that is displayed only the first time the application is run to ask for a username, which should be the ID used to identify each user.
    - One fragment shows the list of all the previous conversations between you and other users ordered by time, most recent message first.
        - If you click in a conversation it will open in a new fragment.
        - If you long-click a conversation, it will pop a dialog box to ask if you want to erase the conversation.
        - In the AppBar you have two buttons, one to create a new conversation which will ask for the username you want to contact and the second button which will open the fragment for the Arduino configurations.
    - One fragment shows a chat between the user and one of its contacts.
        - It should be visible which one sent which message using a similar design as common messaging applications (SMS, WhatsApp, Telegram,...).
        - Each message should also display the time it was sent or received.
        - The AppBar should display the name of the user of the open conversation.

  - One fragment shows the configurations of the Arduino where you can select from all the users you contacted the ones that will fire a notification to Arduino.
- When a new message is received from the selected contacts, the Arduino should display the sender, and the content of the message and also blink a LED to warn the user of the received message.
- **All information must be transmitted using only the MQTT connection**. The transmitted messages must guarantee reliable communication between users. Therefore, they can use any MQTT topic, while the Arduino must subscribe to the topic of the conversation to display the sender's and message's information.
- All the messages should also be stored in the device (it is recommended to use SQLite) allowing the user to see previous conversations.
- When the user receives a new message the application must issue a notification displaying the sender and the content of the message (only when the application is open; you don't need to have a service or something similar for when the app is closed).
- You can use the MQTT server available at: tcp://broker.hivemq.com:1883

Evaluation Criteria

Description	Value
Fragment 1	
Text input validation	5%
Display only one time	5%
Fragment 2	
List of conversations ordered by time	10%
Click and long press	5%
Correct use of the AppBar	5%
Fragment 3	
Display all the messages between the two users	10%
Correct use of the AppBar	5%
Fragment 4	
Filter the users of Arduino notifications	5%
Arduino	
Notification from selected users	10%
LED blink	5%
Communication using MQTT	10%
Correct use of Storage to save the exchanged messages	10%
Use of notifications from new messages	10%
Use of Dynamic Fragments	5%

Deadlines and Evaluation

Milestone 3 (3pts)

- The deadline is on December 17th.