-Highlight what tasks were done during this cycle (by owner). They will come from your Kanban board or tracking tool. Recall that there is an expectation of on-average, equal contribution to the project.

Link to our Trello board:
https://trello.com/invite/b/HqdSQki6/ATTI5e4b8ea385bbca9b506227e79c537255723E5F60/kanban-dev-board

- Created an *mqtt* branch with the following:
    - Examples of sending and receiving messages between many users using the MQTT and Mosquitto technologies, to give an idea on how much we will need to design and program (e.g. we found out we will not need to program an entire multithreaded server for handling requests) as well as to make implementation easier as we'll already know how to interact with the API
    - Examples of connecting to our database (firebase) and using its API to save messages as well as querying all messages after a certain timeframe (which will be handy for offline message retrieval)
    - An example of connecting both the firebase API with MQTT to automatically save any incoming messages to the database
- Made updates to the architecture diagram to describe how our system will work with firebase and MQTT
- Reorganised the repo to put all the diagrams together
- Updated the *possible-structure* branch with the following:
    - Implemented a command design pattern, which allows us to chain complex behaviours in a low-coupling fashion that is both reusable and scalable; it could be used as a way to either call particular functions across the program or act as a way to connect some UI elements directly with low coupling.
- Created a *UI* branch with the following:
    - Created a "main" file as the driver to the UI part of the application (main file describes what screen to show and who's logged in)
    - Created the login interface using email and password
    - Created forgot password interface with an email address field and email to change the password
    - Created a sign-up interface with a username, email and password fields
    - Implemented navigation between the various screens and stack handling (popping/pushing the screens)
- Created the *front-end-base* branch for working on front-end features. Worked on the following:
    - Created skeleton python files from our existing class diagram with empty functions
    - Used pytest to make unit tests for the functions we knew how we were going to implement (still not sure on interacting with the database)
    - Experimented with firebase authentication and successfully implemented login and register functionality (using the "pyrebase" library wrapper)
    - Changed the class diagram to incorporate authentication with firebase
    - Incrementally edited small details as the implementation for things because more clear
    - Working on connecting the UI with the authentication code and other classes

-Summary of the progress

We've made steady progress and almost finished designing the system and started to break into the implementation phase. We have not yet gotten to do a lot of testing in our implementations so far but the key point is that we now have a better idea of what we need to program as well as what technologies we will be using. We also have made progress in the following areas:

- creating the UI (Login, create user, forgot password)
- user authentication
- Database querying and saving messages
- Sending messages
- Keeping diagrams synchronized
- Skeleton classes corresponding with class diagram with tests
- File structure

-Comments on the process;

We are satisfied with our current software development process using Kanban. The use of Kanban has helped our team to work more efficiently and effectively, and resulted in better productivity. One of the advantages for our group is the visual representation of the workflow, which helps us to physically see the flow of work, more easily allowing us to optimise our process. Additionally, the flexibility of Kanban allows us to adapt to changing requirements or priorities, making sure that we can deliver our milestones on time. The use of Kanban boards and WIP limits has helped us to manage our workload and improve our focus, leading to completing tasks faster. We believe that Kanban has been important for our success, and we plan to continue using it as our primary software development process.

-Branches/Tasks completed and tested/merged:

Right now, we have only merged the *mqtt* branch, as many of the branches are currently still in a critical phase where they are unable to be merged just yet, plus some of them have conflicting file structures, which we will be resolving with haste.

For the rest of the tasks completed, see the list at the top of the document.

-If you have a release candidate (in the works)

We currently do not have a release candidate ready.

-Update your testing report so you know what is working. What's working; what's passing; what's not?

What's passing	**What's not passing**

MQTT message sending using sub/pub	All currently implemented UI features (login, forgot password, signup)
Firebase logging in and registering	All of the low-priority functions in client classes
	Command Design Pattern