FIT4004 Assignment 3 Report

During the process of making the program, our group decided to split the problem into distinct parts so that each part could be tested easily without the need to depend on another part. The process of writing tests was mostly done by first writing the functional code and then using unit tests to confirm the logic.

What bugs, if any, did you find with your unit tests?
The only bugs that i found when testings were simple python errors (such as using the wrong variable name) but the tests did help me fix a massive logic error in dealing with how tweets were fetched, before the program used to go through the tweets and check if they were after the end date and if they were then it would terminate the fetching, but that gave an error in the test and it took a bit to figure out why.

Did you think your testing strategy resulted in adequate tests? Please explain why or why not.
Our testing strategy was simply aimed at the functional correctness of the classes as per the spec, this made it easier to write distinct test cases as we already knew what was to be expected. This also made it easier to find ways in which we could try and alter the system so the functionality could break and thus wrote tests for those as well.

Do you think it works better for the programmer responsible for code to write the unit tests for it, or for somebody else to do it? Explain why.
I feel it is better for the programmer responsible for the code to write the unit test for the given code, however it is up to them to write code which can be clearly understood by others so that in the future they may also go and tweak/add test cases if they do so wish. My reasoning is that when the person responsible for the code writes the unit test, it also reveals their way of thinking in the given scenario which can sometimes be hard to pick up just by looking at comments. It also works better because that person knows exactly what they need to test whereas if they handed it to someone else then it may take them time to find tests cases which cover the complete logic of the code.

How long did writing mock code take? Was it a major component of the overall effort?
Not very long, reading about mock and looking at examples was the simplest way to understand how to use it. Once that was figured out it was simple as writing any other piece of unit test code.

What would you do differently next time if faced with a similar task?
Maybe change the testing strategy to see if it works better than what we used in this project but apart from that nothing would be that much different.