# Distributed Systems

# Assignment 2

# Changes


| | |
|---|---|
| Filename | readme.txt |
| Author | Songzhe Li |
| Student ID | a1767109 |
| E-mail | a1767109@student.adelaide.edu.au |

## 1 Previous Thoughts and Feedback of Draft

- For Assignment 2 Draft I only implemented a simple server and client model, and made the following plan:

* Briefing

I plan to apply a producer-consumer pattern on this project:

The producer takes requests from the ATOM server, push the server sockets to the Priority Blocking Queue (which is a buffered area in this pattern).

Each server socket stores the data from GETClient or the Content Server

The consumer takes the sockets from the queue

At this stage, I have only implemented the aggregation server initiation.

In the aggregation server, the lamport clock will be initialized at 0 and it ticks every time when the producer (consumer) receives a request

There would also be a XML parser to translate the message between the ATOM server and the content server which will be implemented later

- Draft Feedback

**Feedback** posted(Wed 22 Sep 2021 19:07:31)

Extra Marks: 0
1. Does the client communicate with the Aggregation Server?
No
2. Does the Content Server communicate with the Aggregation Server?
No
3. Does the basic control flow work?
No
4. Does the aggregation server order requests?
No
5. Feedback

6. Is XML parsing implemented?
No
7. Feedback

8. Is the Aggregation Server persistent?
No
10. Feedback

11. Is restore on failure implemented?
No
12. Feedback

13. Are Lamport clocks implemented?
No
14. Feedback
No
15. Is the heartbeat implemented on the Aggregation Server?

16. Is the heartbeat implemented on the Content Server?
No
17. Feedback

18. Is the system fault tolerant?
No
19. Improvements
No
20. Is testing present?

21. Is testing comprehensive?
Almost
22. Feedback
You submitted an early work in progress. Please make sure you comment your code when submitting the final version.
23. Does the code follow the code checklist?
Almost
25. Is the design of good quality?

26. What improvements need to be done to improve the design quality?
I can't wait you to submit the final version. Please make sure you detail everything in the readme file and and make sure that everything written in there can be followed. Good luck

## 2 Changes Compared to Draft

According the above plan and feedback, I have made the following changes:

1. Instead of having a separate class of Lamport Clock, I integrated the clock in the Aggregation Server, so that I can easily access and increment the clock.

2. For a clearer logic, I moved the class Consumer and Producer into the Aggregation Server class. This helped me to have a clearer logic when implementing. Meanwhile, this also renders less java files in a folder, making the folder cleaner.

3. I replaced the Priority Blocking Queue with Blocking Queue, it can suit better in the producer-consumer model in java.

4. For testing part, in the draft, I could only start and shut down server. However, in my final version, I have developed four methods to test my program:

[1] single client and server -> connection of single client and server

[2] single client timeout -> which will delete content after 12s not responding

[3] multiple client and server -> connection of multiple client and server

[4] multiple clients and server timeout -> which will delete content after 12s not responding (multiple clients and servers version)