

COMP6231/1 – Sections BB – Summer 2018

PROJECT

Due Date: Sunday, July 29th by midnight 11:59 PM

Important Note

- The work is realized in a **team of four (04) students**.
- Presentation of the project is scheduled on **Saturday, July 28th, 2018**
- Submit the documentation and the code by the due date.
- Your program should be compiled, executed and return the expected results; otherwise a mark 0 (zero) will be assigned.

Software Failure Tolerant/Highly Available Distributed Class Management System (DCMS)

In this project, you are going to enhance your CORBA implementation of the Distributed Class Management System (DCMS) developed in Assignment #2 to be software failure tolerant or highly available using process replication. This project is suitable *for teams of 4 students*, as described in the following:

In this project, you are going to implement a highly available CORBA Distributed Class Management System, which tolerates process crashes only (no software bugs) using unreliable failure detection. Thus, there is a group of (at least 3) server processes (typically running on different hosts) providing the redundancy for high availability and periodically checking each other for failure detection. One of the processes in the group is the elected leader and receives requests from clients through a CORBA front end and sends responses back to them through the front end. The leader of the server group broadcasts the client request atomically to all the servers in the group using a reliable FIFO broadcast mechanism, receives the responses from them and sends a single response back to the client as soon as possible. Since the replicated servers are usually on a local area network, they communicate using the unreliable UDP protocol. However, the communication among them should be reliable and FIFO. Specifically do the following

- Assuming that processor failures are benign (i.e. crash failures) and not Byzantine, design your highly available active replication scheme using process group replication and reliable group communication.
- Design and implement the group leader process which receives a request from the front end, FIFO broadcasts the request to all the server replicas in the group using UDP datagrams, receives the responses from the server replicas and sends a single response back to the front end as soon as possible.

- Design and implement a reliable FIFO broadcast subsystem over the unreliable UDP layer.
- Design and implement a failure detection subsystem in which the processes in the group periodically check each other and remove a failed process from the group. If the group leader has failed, a new leader is elected using a distributed election subsystem.
- Design and implement a distributed leader election subsystem (based on the bully algorithm), which will be called when the current leader has crashed to elect a new leader for the process group.
- Modify the individual implementations of the server replica from *Assignment #2*, integrate all the modules properly, deploy your application on a local area network, and test the correct operation of your application using properly designed test runs. You may simulate a process crash by killing that process while the application is running.

MARKING SCHEME

[10%] Design Documentation.

- [5%] Describe and explain your design and architecture clearly, including theories (protocol, algorithm) you apply, how to implement, and also describe dataflow (how your modules interact / cooperate with each other to achieve the function).
- [5%] Design proper and sufficient test scenarios, which should include testing data and results.

[5%] Group Demo in the lab or in the office on **Saturday, July 28th, 2018**.

EDUCATIONAL GUIDELINES

- The work is realized in a team of four (04) students.
- The delivery (document and source code) must be made no later than **Sunday, July 29th, 2018 by midnight 11:59 PM** using the Website submission at <https://fis.encs.concordia.ca/eas>.
- If you are having difficulties understanding sections of this assignment, feel free to email the Teaching Assistants. It is strongly recommended that you attend the tutorial sessions which will cover various aspects of the assignment.