How to Use this repository

Prerequisite

1. Install Docker

Steps to Run

1. In the docker CLI change to directory to the path where docker-compose.yml along with other folders are present execute $ **docker compose build**
2. Execute $ **docker compose up**
3. Wait for some time to allow system to become stable and leader got selected
4. Access the Controller CLI from docker desktop
5. Sample commands that can be used to test the system

```
>>python Controller_request.py Node1 LEADER_INFO
```

– This will send request to Node1 to request for LEADER_INFO. Node1 will return LEADER_INFO to the controller back which is printed as result in the Controller CLI.

```
>>python Controller_request.py Node2 TIMEOUT
```

– This will send request to Node2 to timeout immediately

```
>>python Controller_request.py Node2 SHUTDOWN
```

– This will send request to Node2 for shutting it down Node2 will exit without throwing any error.

```
>>python Controller_request.py Node3 CONVERT_FOLLOWER
```

– This will send request to Node3. If Node 3 is leader, then it will become follower and stop sending heartbeats

>>**python Controller_request.py Node3 RETRIEVE** – This will send request to Node3. If Node 3 is leader, then it will reply with log entries and if it is follower then it will reply the LEADER_INFO.

>>**python Controller_request.py Node3 STORE key1 Value1** - This will send request to Node3. If Node 3 is leader, then it will add key1 and value1 to its log along with the term and if it is follower then it will reply the LEADER_INFO.

Access the Leader UI from the browser using the URL

- If Node1 is Leader- http://localhost:8095/student-course-reg/
- If Node2 is Leader- http://localhost:8096/student-course-reg/
- If Node3 is Leader- http://localhost:8097/student-course-reg/
- If Node4 is Leader- http://localhost:8098/student-course-reg/
- If Node5 is Leader- http://localhost:8099/student-course-reg/

Hit the below end points (exposed for verification) in the followers' servers to check data replication

- If Node 1 is Follower - http://localhost:8095/student-course-reg/student/getCourse
- If Node 2 is Follower - http://localhost:8096/student-course-reg/student/getCourse
- If Node 3 is Follower - http://localhost:8097/student-course-reg/student/getCourse
- If Node 4 is Follower - http://localhost:8098/student-course-reg/student/getCourse
- If Node 5 is Follower - http://localhost:8099/student-course-reg/student/getCourse

6. Inside the container check the below directories for text file where the data is getting saved
   - cd /data/info
   - cat userInput.txt
   - cat nodeinfo.json

7. In the host I created five different volume mount to persist the data:
   - /nodeinfo1
   - /nodeinfo2
   - /nodeinfo3
   - /nodeinfo4
   - /nodeinfo5