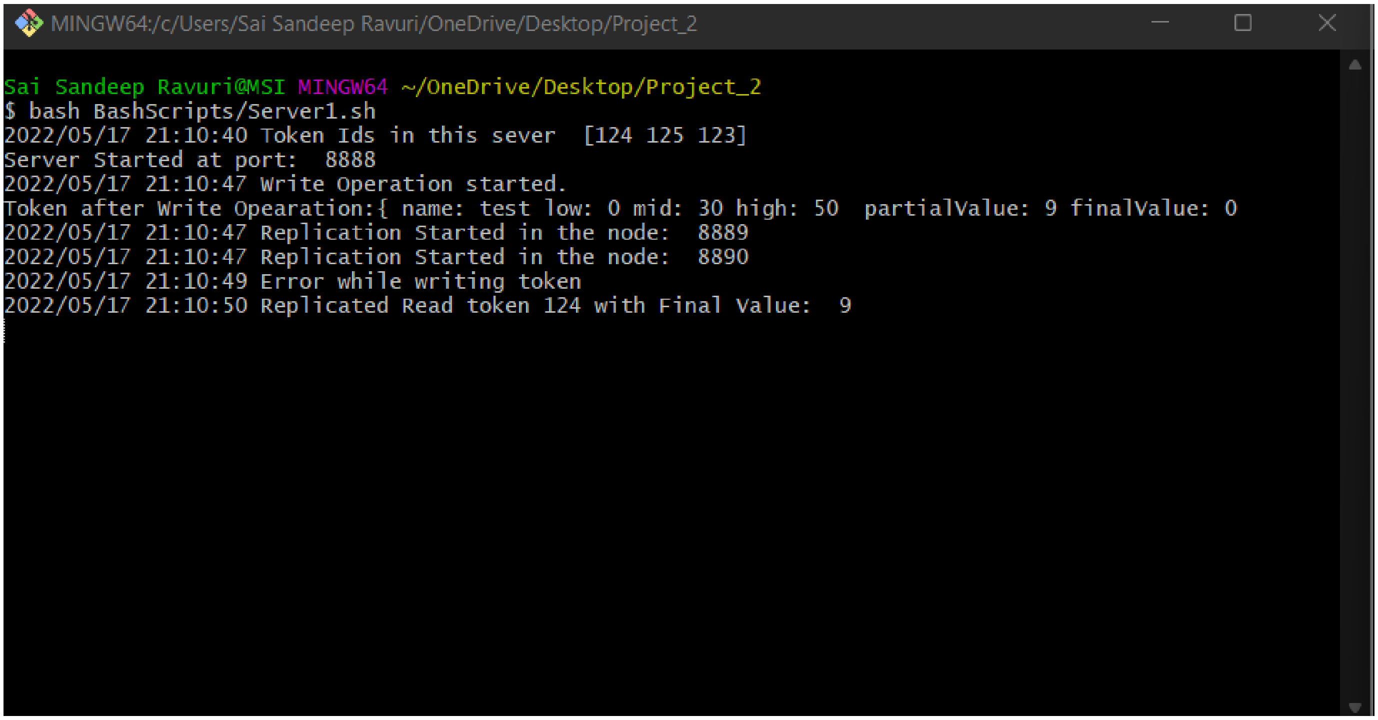
References:

I have referred from the links given in the problem statement pdf itself and watched few you tube videos to know the about the implementation of Mutex. Below are the listed references that I read tobuild the project

1. https://grpc.io/docs/languages/go/quickstart/
2. https://developers.google.com/protocol-buffers/docs/gotutorial
   From the above links I installed grpc in my laptop and created the proto files. Followed the stepsand created the basic connection and sent a message through the client.
3. https://pkg.go.dev/google.golang.org/grpc#Server
   From the above link I have got the gist of implementing different functions like create, delete,write, and read
4. https://gobyexample.com/command-line-flags
   Here I used this example to build the command line flags for the client and server files
5. https://www.youtube.com/watch?v=QmIdWTidEa8
   From this youtube video I referred the functionality of the proto, client, and server files
6. https://www.youtube.com/watch?v=JlmYLPxw
   VzQFor mutex implementation
7. To convert json to yaml
   https://onlineyamltools.com/convert-json-to-yaml
8. To Read the yaml file
   https://zetcode.com/golang/yaml/
   https://stackoverflow.com/questions/30947534/how-to-read-a-yaml-file
9. Atomic Semantics & Write all Read one quorum
   https://www.youtube.com/watch?v=uNxl3BFcKSA&t=239s
10. Random integer generation -
    https://www.geeksforgeeks.org/generating-random-numbers-in-golang/

Output Screens:

1. Running the Server1 bash script file

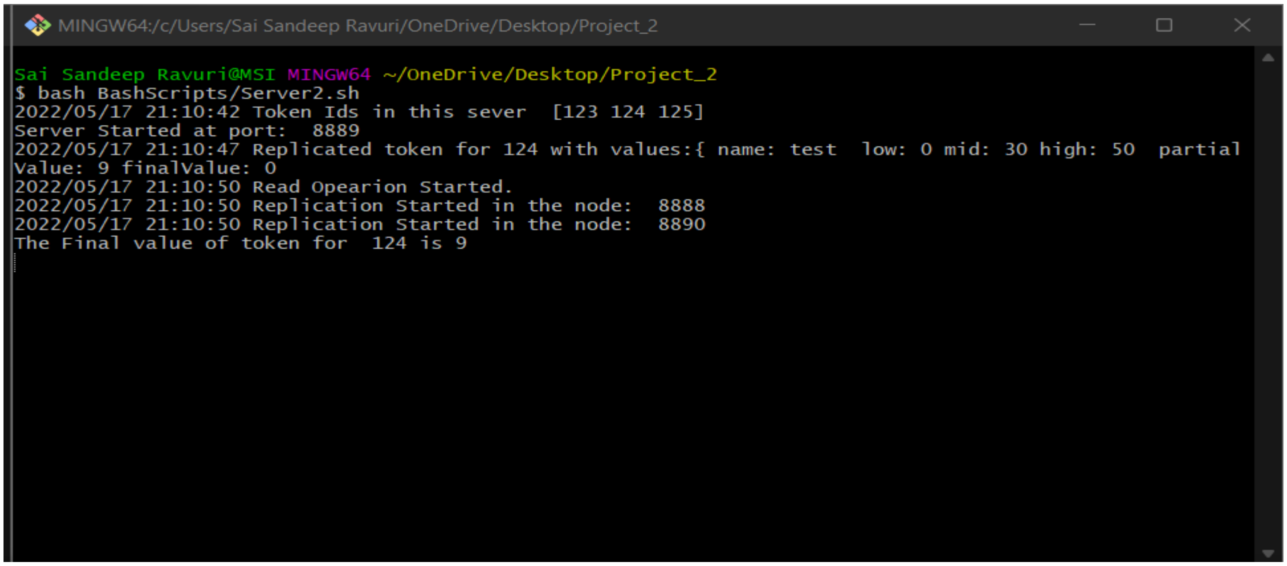


MINGW64:/c/Users/Sai Sandeep Ravuri/OneDrive/Desktop/Project_2

```
Sai Sandeep Ravuri@MSI MINGW64 ~/OneDrive/Desktop/Project_2
$ bash BashScripts/Server1.sh
2022/05/17 21:10:40 Token Ids in this sever  [124 125 123]
Server Started at port:  8888
2022/05/17 21:10:47 Write Operation started.
Token after Write Opearation:{ name: test low: 0 mid: 30 high: 50  partialValue: 9 finalValue: 0
2022/05/17 21:10:47 Replication Started in the node:  8889
2022/05/17 21:10:47 Replication Started in the node:  8890
2022/05/17 21:10:49 Error while writing token
2022/05/17 21:10:50 Replicated Read token 124 with Final Value:  9
```

Server1 starts with 8888 localhost and displays all the keys on this server. Then a request is made from a client with a particular token id. This Token Id's writer node is already known to the client and server nodes through the YAML file. After the writer node is retrieved from the YAML file, which in this scenario is '8888'. So, this port starts the Write Operation and replicates it to the other servers which are '8889','8890'.

2. Running the Server2 bash script file

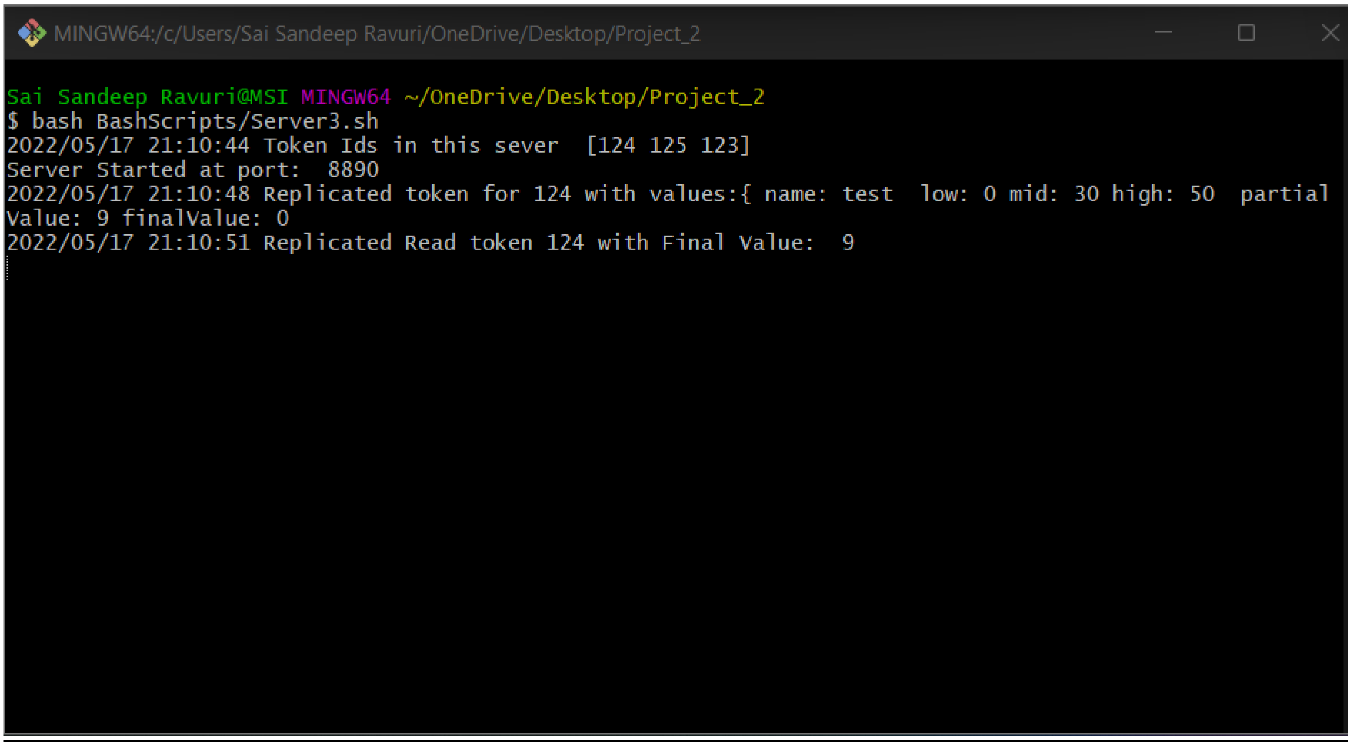


MINGW64:/c/Users/Sai Sandeep Ravuri/OneDrive/Desktop/Project_2

```
Sai Sandeep Ravuri@MSI MINGW64 ~/OneDrive/Desktop/Project_2
$ bash BashScripts/Server2.sh
2022/05/17 21:10:42 Token Ids in this sever  [123 124 125]
Server Started at port:  8889
2022/05/17 21:10:47 Replicated token for 124 with values:{ name: test  low: 0 mid: 30 high: 50  partial
Value: 9 finalValue: 0
2022/05/17 21:10:50 Read Opearion Started.
2022/05/17 21:10:50 Replication Started in the node:  8888
2022/05/17 21:10:50 Replication Started in the node:  8890
The Final value of token for  124 is 9
```

The request from '8888' for replication processes to this server2 and replicates the token. And afterwards a read operation is performed by this server by retrieving the reader nodes from the YAML file and replicating it in other servers.

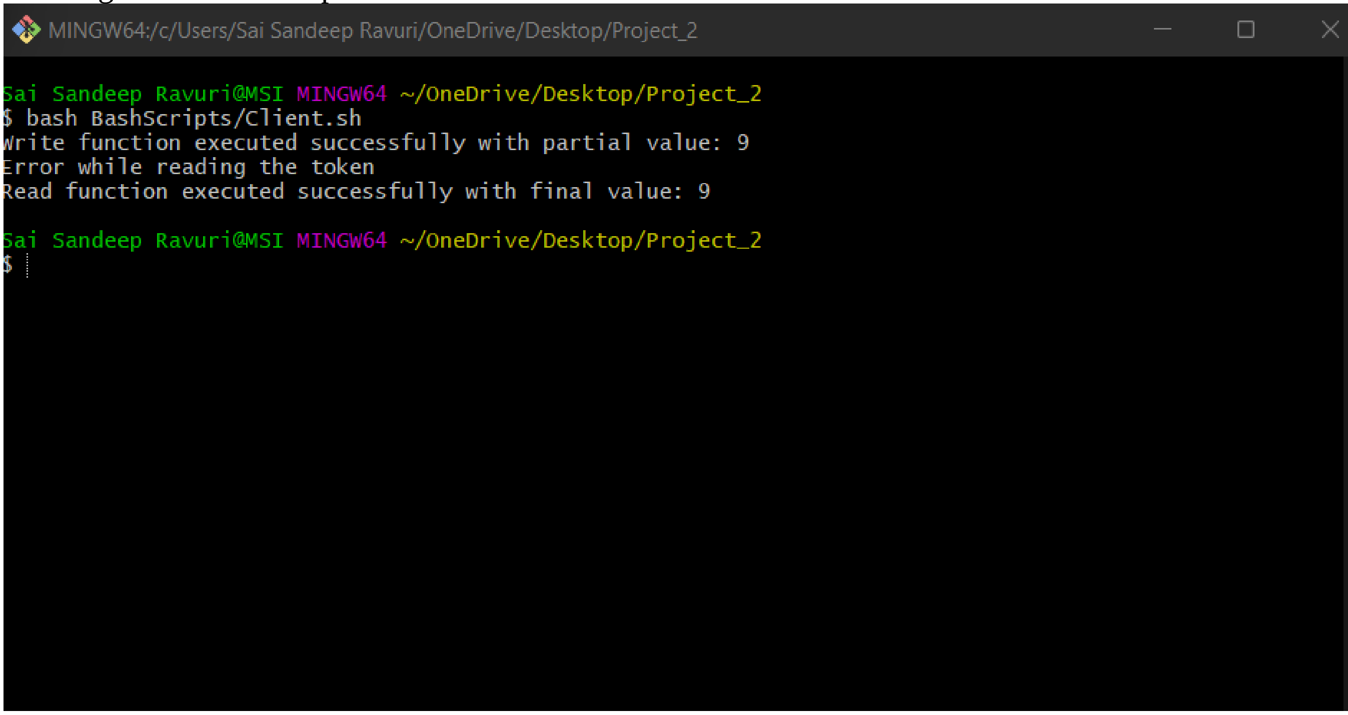
3. Running the Server3 bash script file


MINGW64:/c/Users/Sai Sandeep Ravuri/OneDrive/Desktop/Project_2

```
Sai Sandeep Ravuri@MSI MINGW64 ~/OneDrive/Desktop/Project_2
$ bash BashScripts/Server3.sh
2022/05/17 21:10:44 Token Ids in this sever  [124 125 123]
Server Started at port:  8890
2022/05/17 21:10:48 Replicated token for 124 with values:{ name: test  low: 0 mid: 30 high: 50  partial
Value: 9 finalValue: 0
2022/05/17 21:10:51 Replicated Read token 124 with Final Value:  9
```

The Replicated Values for write and read operations are reflected here.

4. Running Client bash script file



MINGW64:/c/Users/Sai Sandeep Ravuri/OneDrive/Desktop/Project_2

```
Sai Sandeep Ravuri@MSI MINGW64 ~/OneDrive/Desktop/Project_2
$ bash BashScripts/Client.sh
Write function executed successfully with partial value: 9
Error while reading the token
Read function executed successfully with final value: 9

Sai Sandeep Ravuri@MSI MINGW64 ~/OneDrive/Desktop/Project_2
$
```

The response from all the servers are displayed here, only if successful. Also, there is a message “Error while reading the token” in the client side, which discussess about the fail silent model.