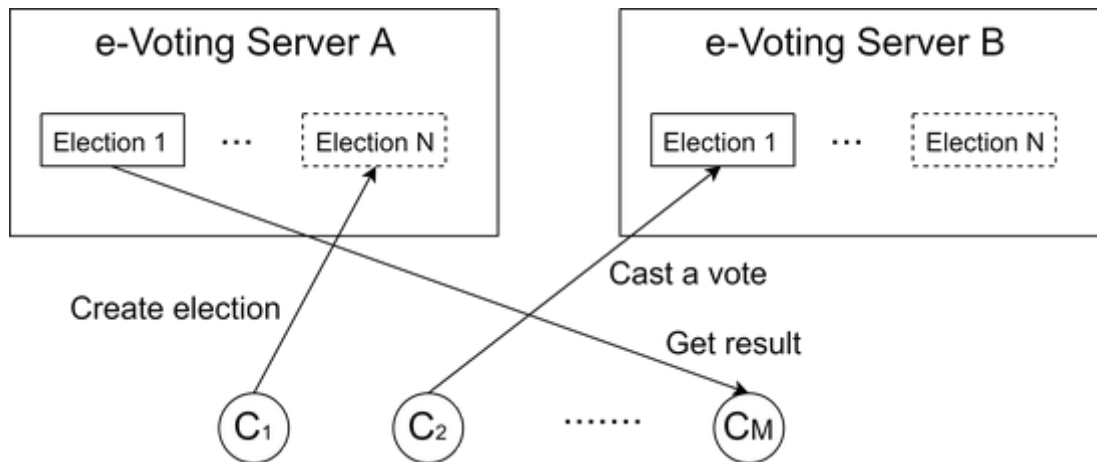


(FTC) Active-Active e-Voting Server

In this project, we will implement active-active redundancy for the e-Voting service. The service consists of two servers, A and B, as shown in Fig. 1. The clients can contact any server to perform the voting operations (Fig. 2).



```
service eVoting {  
    rpc PreAuth (VoterName) returns (Challenge);  
    rpc Auth (AuthRequest) returns (AuthToken);  
    rpc CreateElection (Election) returns (Status);  
    rpc CastVote (Vote) returns (Status);  
    rpc GetResult(ElectionName) returns (ElectionResult);  
}
```

Fig. 2. RPC APIs for e-Voting service

Failure Mode

You only need to consider network partition and node crash.

When there is no failure, the system should provide all the features of the e-Voting service (Fig. 2).

When failure occurs, it is ok to suspend the e-Voting service.

When the failure is cleared, the system should resume all the service features.

Submission Checklists

1. Your code on GitHub
2. Prepare a README describing how to build and run your code
3. Write a report that covers
 - a. The design and implementation of your active-active e-Voting server
 - b. The evaluation of your e-Voting server and client. You need to demonstrate that the servers can deal with the failures listed in the previous section.