Lab 9: Real Time Elections / Voting with Socket Programming

Objective of this lab:

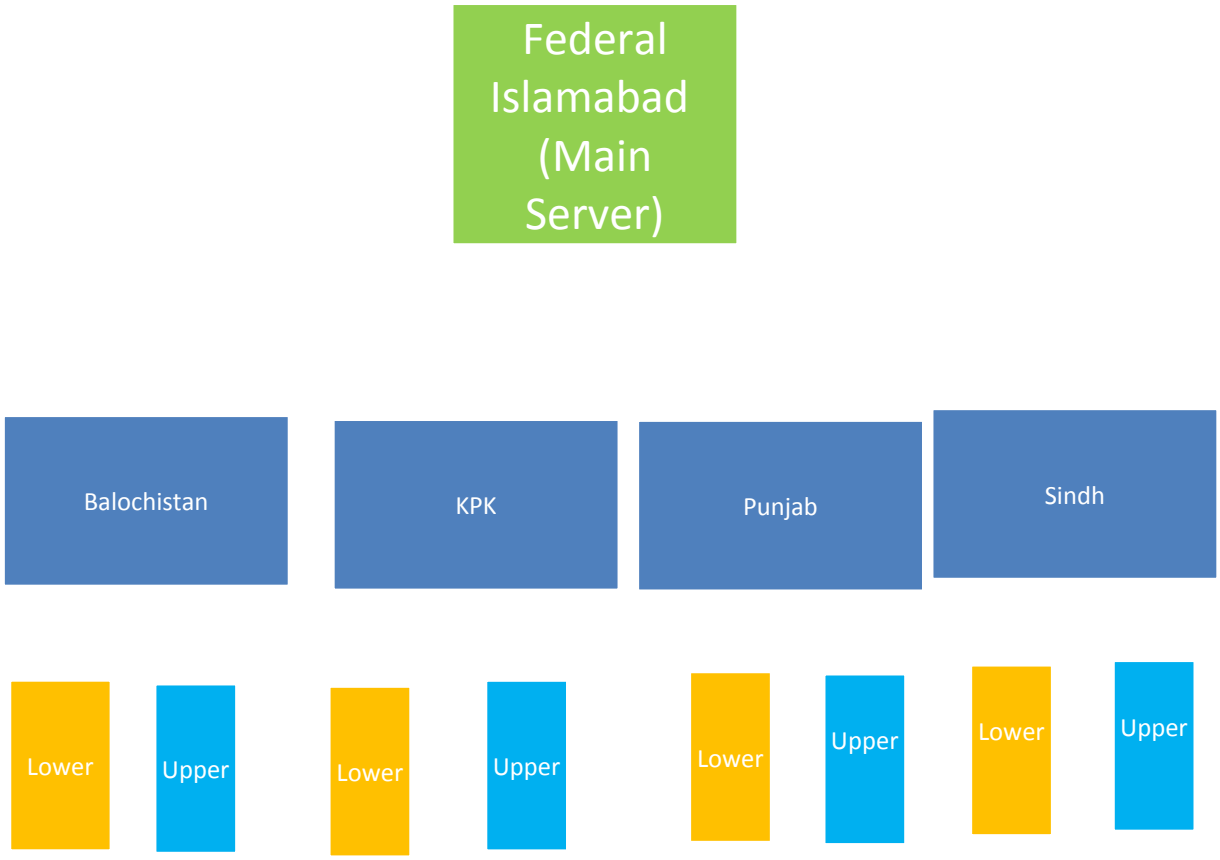
In this lab, we'll investigate

- Simulate real time voting scenario of Pakistan
- Communication between different servers and sub servers in real time
- Transfer of messages forward and backward to particular client(both directions)
- Results update in no time

Instructions:

- Read carefully before starting the lab.
- These exercises are to be done individually.
- Complete t
- At the end ding

Lab Task



Real Time Elections (voting):

We hear the news of rigged/fraud elections every time elections held in Pakistan. In the last elections many people were complaining of casting a vote of one party but it turned out for other party

The idea is very simple, any person can cast a vote, and his /her vote should be updated on main screen (main server) on the spot. It means results are updated in real time, no need to wait for results.

Methodology:

As there are four provinces in Pakistan and each province has south and north regions, for example Punjab is divided into south Punjab (Multan, Dera ghazi khan etc.) and north Punjab (Rawalpindi, Murree etc.)
There is a main server in Federation and it has four sub servers known as provincial (Punjab, Sindh, Baluchistan, and KPK) servers and each provincial server has two regional (South and

North) sub servers. So, each province has a dedicated server and two sub servers to handle regions (South and North)

People will cast their vote only at regional level of their province (Upper/Lower) and their vote will be updated on provincial server as well as main server in real time.
Following queries should be entertained/displayed all the time on provincial servers and main server.
For simplicity assume there are only two parties.
PMLN and PTI

Total number of votes of PTI and PMLN: This result is updated every time vote casted.

Casting can be stopped at sub server level anytime.

At the end of the election Main Server should announce the winner based on number of Votes.