CS 433
IIT Gandhinagar

Mini-LinkedIn

Project Presentation

Project Goals

Using networking paradigms to design Mini-LinkedIn tool

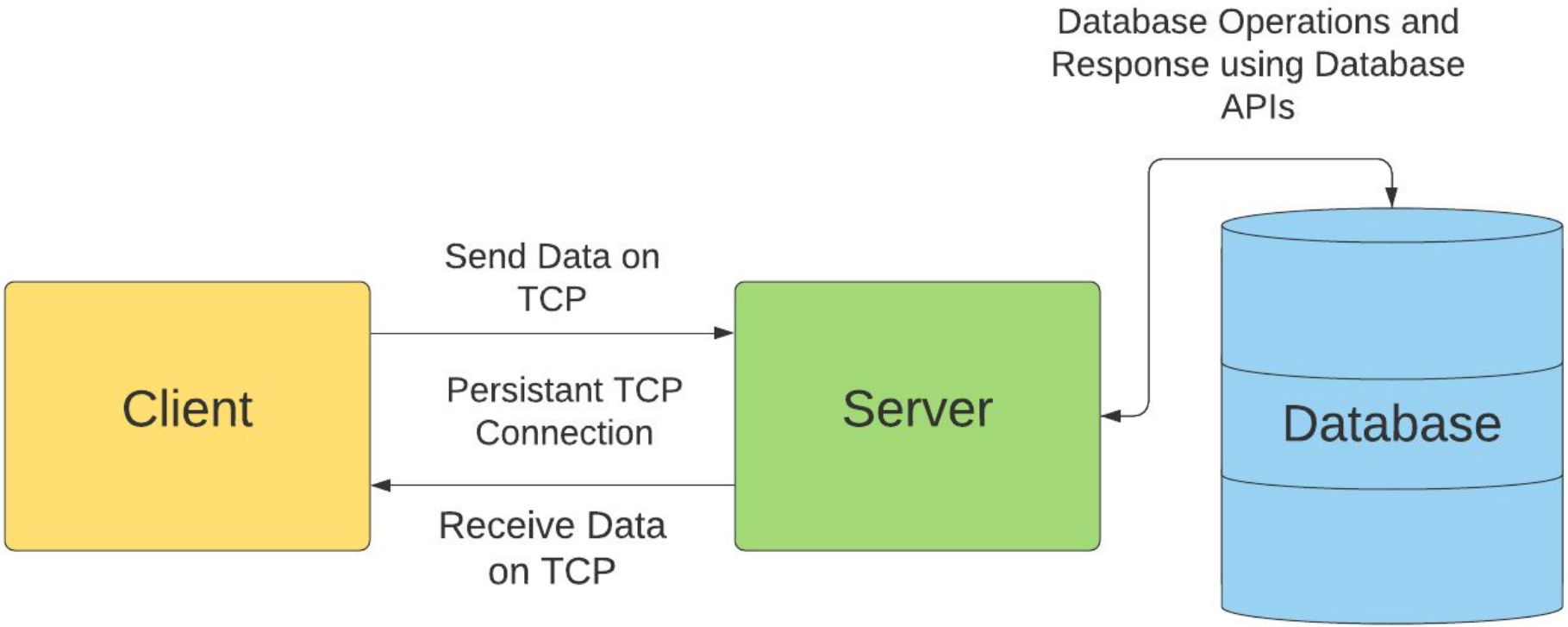
Automated testing procedure on Mininet

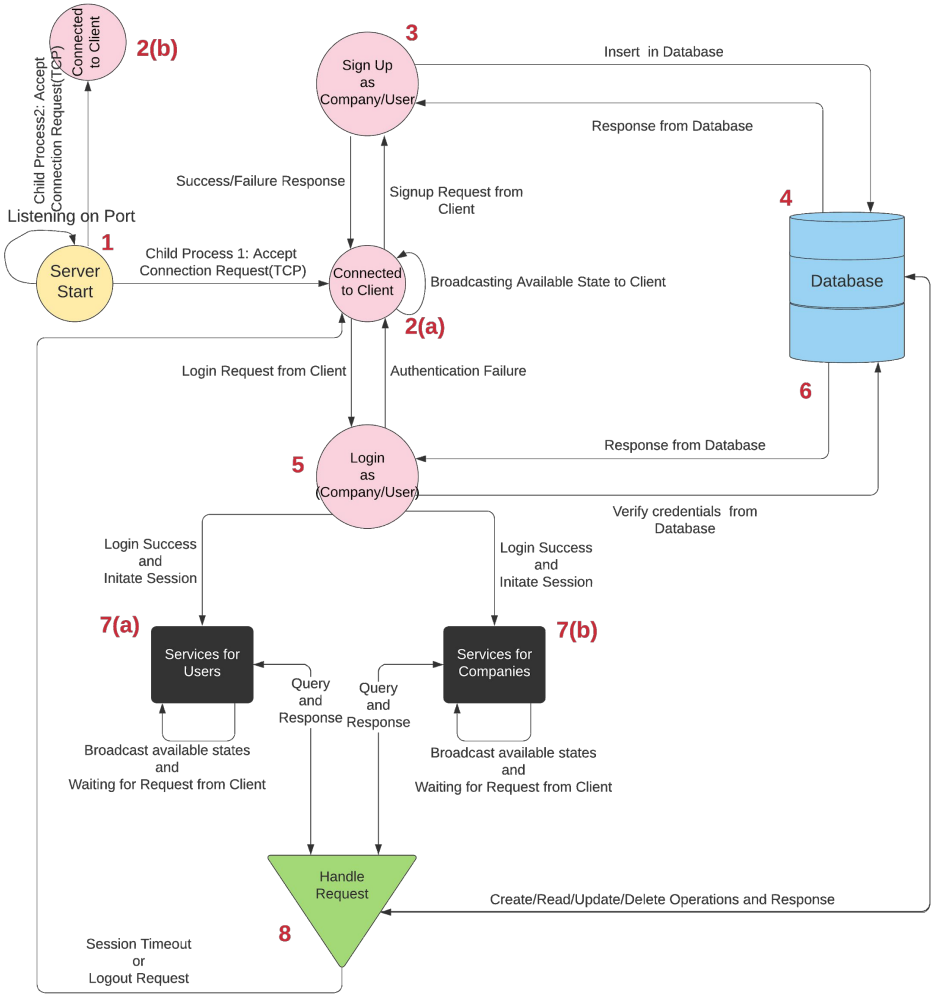
Aditya Tripathi 18110010
Kushagra Sharma 18110091
Nishikant Parmar 18110108

Thanks to Prof. Sameer G Kulkarni

High Level Design (HLD)

Client-Server Model

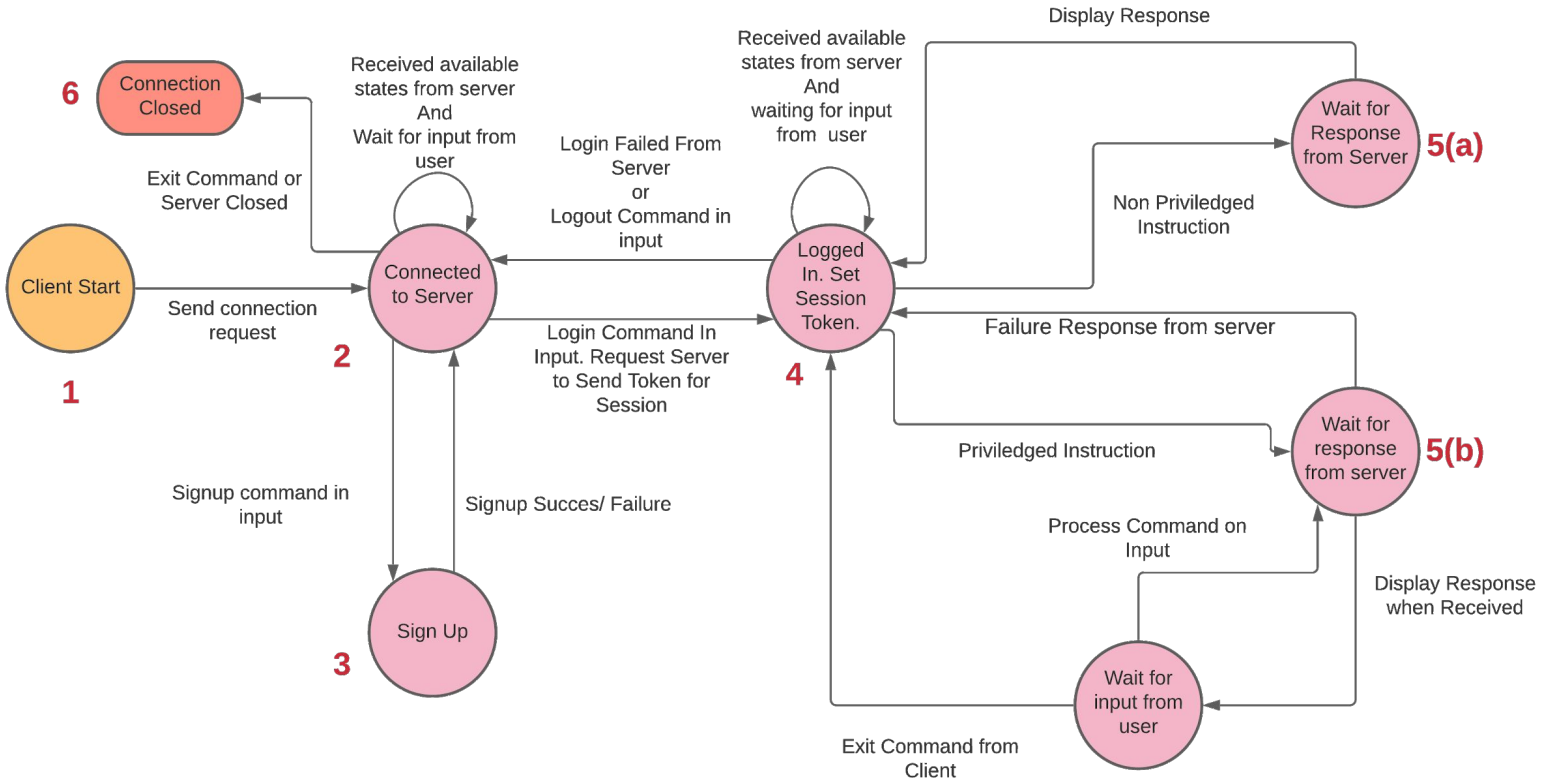


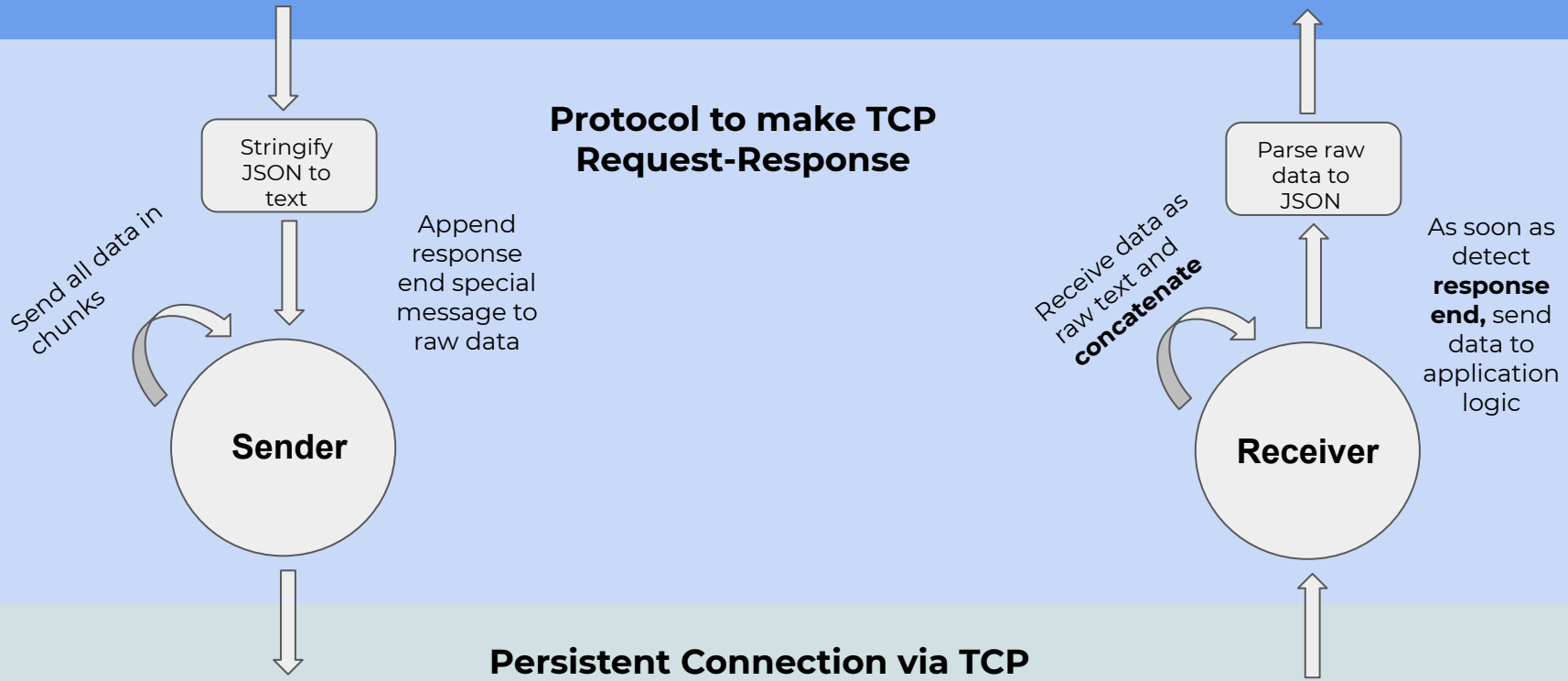


High Level Design (HLD)

Server's Finite State Machine (FSM)

High Level Design (HLD), Client's Finite State Machine (FSM)




Application Logic
Stringify JSON to text
Send all data in chunks
Append response end special message to raw data
Sender
Protocol to make TCP Request-Response
Parse raw data to JSON
Receive data as raw text and concatenate
As soon as detect response end, send data to application logic
Receiver
Persistent Connection via TCP Socket

Protocol Design

The JSON structure is as follows -

Client to Server -

```
{
  'command' : <type:String, instruction/route for server>,
  'body' : <type:JSON, data for server>,
  'token' : <type:String, client's token for session management>
}
```

Server to Client -

```
{
  'status': <type:String, status code of request handled by server>,
  'message' : <type:String, message corresponding to the status code>,
  'data' : <type:JSON, payload after handling request>
}
```

Feature Checklist

- Login and signup flow for a user and a company.
- Users can post and react to their connection’s posts.
- Users can send connection requests to other users on Mini-Linkedin.
- Users can edit and create their profile where they can add skills, experience etc.
- Users can accept connection requests from other users.
- Users can view the people who have viewed their profile.
- Users can endorse skills of their connections. And users can also see the people who have endorsed their skills.

Feature Checklist

- A company will be able to post jobs, and view the profile of users who have applied to the job.
- Users can search for jobs based on their skills as filters. The search query will be made on all the available jobs on the Mini-LinkedIn platform.
- The server is scalable and can handle concurrent client requests.
- The password entering field at login/signup is masked to hide it while entering.
- There are multiple reactions available for posts - like, support and clap
- Users can comment on any posts in their feed

Low Level Design (LLD)

Implementation

Node.js is an open-source, cross platform runtime environment for executing JavaScript code outside of a browser. It works on an event based model requiring callback functions which provides better performance than traditional thread based models.

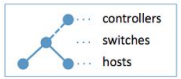
A TCP connection can be easily set up on Node.js using sockets that allows client-server interaction in Mini-LinkedIn.

MongoDB is a document-based NoSQL database with powerful querying capabilities, flexible schemas which suit the requirements of the Mini-LinkedIn tool. It is nicely compatible with Node.js, offers code-native data access (APIs) for the server to work and has enough documentation.

Testing Procedure

Workload Generation

Random workloads are generated using `pickle` and ‘faker` and `random` modules. The data of user and company are stored in the respective workload folders.



Creating Mininet Topology

Created Mininet Tree topology with switches, assigned clients to their servers (load balancing) and ran the server and clients on the host nodes.

Automated Testing on Interactive Client

Testing script `mini_test_user.py` or `mini_test_company.py` read the user or company data and send it to the client which in turn requests the server with the provided data and command and waits for response.

Manual Testing

Created an interactive client on CLI. Users can connect to the server and follow instructions to realize all features of Mini-Linkedin.

Security Aspects

Authentication through token

For every coming request from the client, the server verifies the token and command and then takes necessary action. If the authentication via the token fails the commands will not be handled by the server unless the client logins with proper credentials. The client is not permitted to perform any command except login/sign up without logging in.

Restricting users to user APIs and companies to companies APIs

A user cannot access/use APIs for a company and a company cannot access/use the APIs for a user. For instance, only a user will be able to send connection requests to other users and a company cannot send connection requests. Similarly, a user will not be able to post jobs, only a company will be able to post jobs.

Passwords

The passwords entered by the client are obscured using (*) symbol for security.

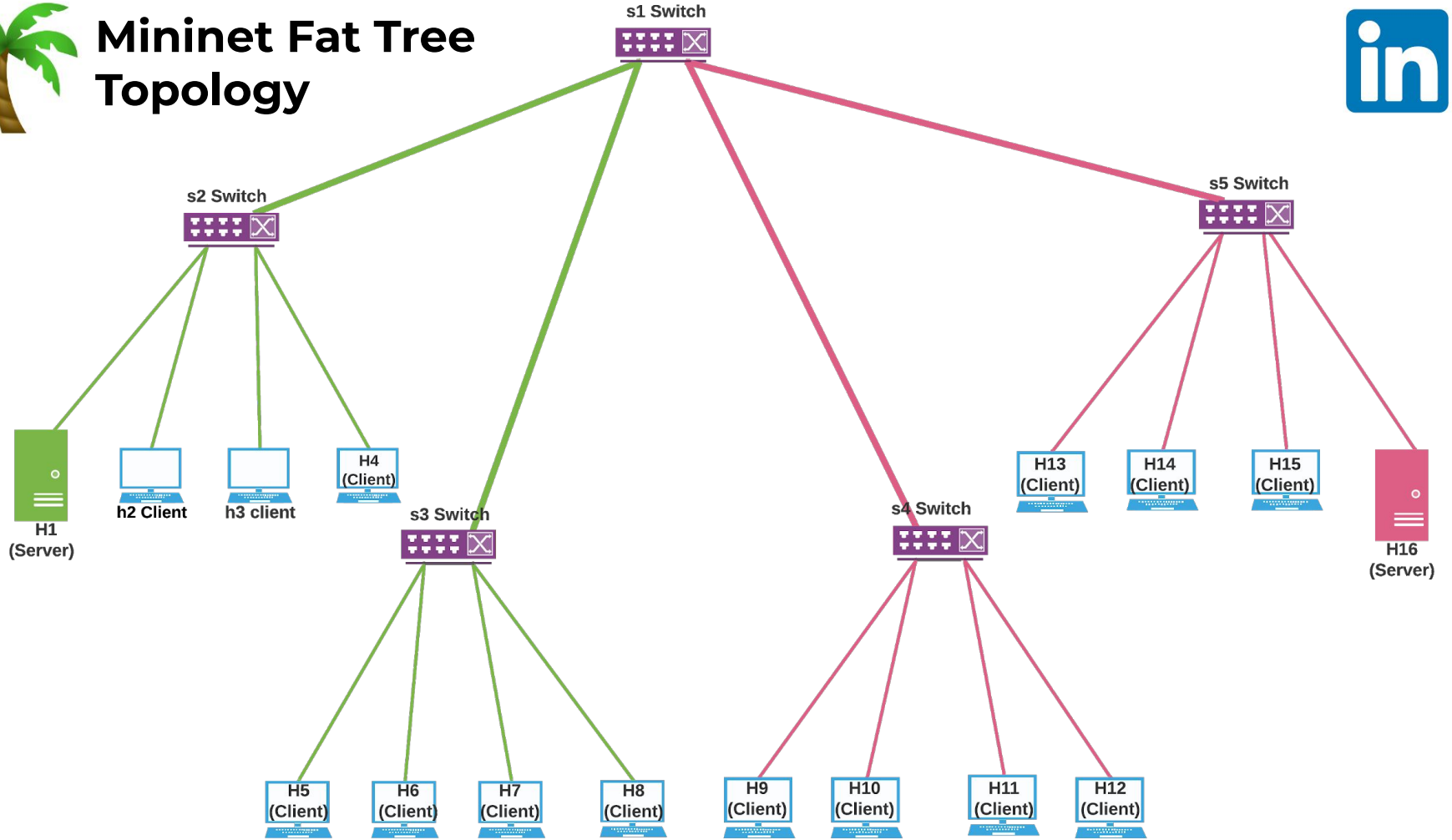
Encoding

All the data sent over TCP connection is encoded before sending.

Feature level security

A user can only see posts of users who are in their connection. Similarly, a user who is already a connection of another user, can't send another connection request. Similarly, a liked post cannot be liked again by the same user.

Mininet Fat Tree Topology



Thank You

References for Logos and Graphic Icons used in slides - Google Images