

Group - 99

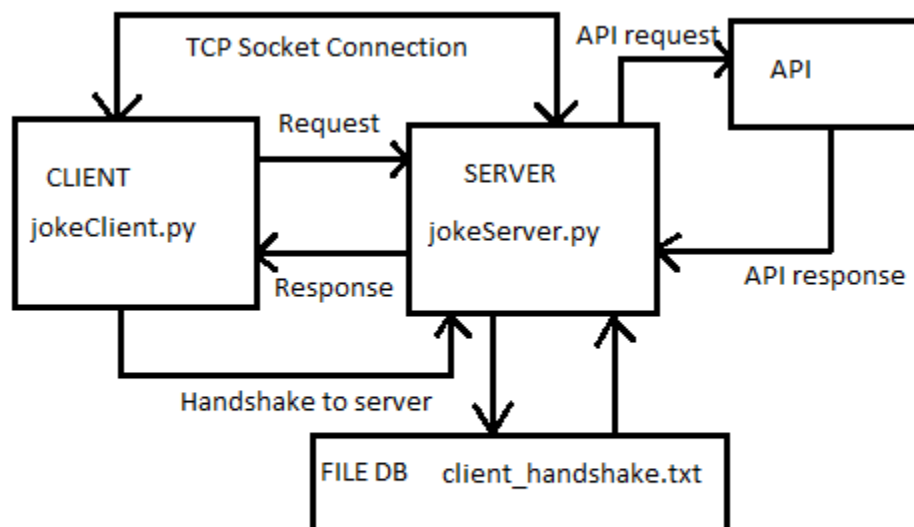
Name - Abhishek kumar, 50419133

Dad Joke Generator

Deployment:

1. Copy zip file
2. Unzip to a location
3. Open cmd/terminal
4. Navigate to project location "dad_joke_generator"
5. Here we will find the "docker-compose.yml" file.
6. Execute commands the below commands in :
 - a. docker-compose build
 - b. docker-compose up
6. Rerun command "b." to view saved messages from the previous run.
7. After output is visible in cmd/terminal run:
 - a. Docker-compose down

Design



Client-Server System Design

Client Server Interaction

- The client and server interact using TCP socket based connection.
- Both client and server are running in local machine
- We create connections using hostname and port number.
- We use socket.AF_INET to use the ipv4 address.
- We use socket.SOCK_STREAM to use connection oriented services
- 0 is for using TCP protocol
- socket.gethostname() to get the ip address of the machine

Interaction :

- When we start the server (jokeServer.py) , it waits for any request from the client.
- After client (jokeClient.py) request is made, server establishes a connection to send and receive messages
- Server makes an API call and fetches a joke (in plain/text format) and sends it to the client.
- After the joke is received by the client, the client views it in the terminal and sends a handshake to the server.
- Server stores this handshake in client_handshake.txt file.
- Server shows the current and previous stored handshakes in the terminal.
- After this the connection is closed.

Project Structure

```
dad_joke_generator
├── docker-compose.yml
├── client
│   ├── jokeClient.py
│   └── Dockerfile
├── server
│   ├── jokeServer.py
│   ├── Dockerfile
│   └── client_handshake.txt
```