CSE312 TCP connections Report

Web Framework: Flask / Python

General Information & Licensing

Code Repository	https://github.com/miguelgrinberg/Flask-SocketIO
License Type	MIT
License Description	● Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: ○ The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
License Restrictions	● THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Magic ⋆★⁺₊°·.°☽°⌒☄。°★彡⁺⋆ ༄

- CSE312_GroupProject-/Server/project_server.py (Line 1005 - 1006)
- Link: https://github.com/XuyangLi-12088/CSE312_GroupProject-/blob/master/Server/project_server.py#L1005

```
if __name__ == '__main__':
    socketio.run(app, host = "0.0.0.0", port = 8080, debug = True)
```

- socketio.run() function will start up the server, and it replaces the app.run() standard Flask development server startup. After the server starts it will create a TCP connection between clients and server.
- socketio.run called "run" function in flask_socketio/__init__.py (Line 553)
- Link: https://github.com/miguelgrinberg/Flask-SocketIO/blob/main/src/flask_socketio/__init__.py#L553
    - This "run" function will run the SocketIo web server
    - Pass "app", "host", "port", "debug" values into the "run" function
        - App is the Flask application instance, the SocketIo web server need to know what application needs to running on
        - Host is the hostname or IP address for the server to listen on. Defaults to 5000
        - Port is the port number for the server to listen on. Defaults to 5000.
        - "Debug = True" means to start the server in debug mode.
    - Check value of "host", "port", and "debug"
    - If no "host" or no "port", set to default value (line 587 - 594)
        - Host: "127.0.0.1"; Port: "5000"
        - Link: https://github.com/miguelgrinberg/Flask-SocketIO/blob/main/src/flask_socketio/__init__.py#L587
    - Check 'async_mode'(line 630)
    - Link: https://github.com/miguelgrinberg/Flask-SocketIO/blob/main/src/flask_socketio/__init__.py#L630
        - Since we are using 'eventlet' (line 653)
        - Link: https://github.com/miguelgrinberg/Flask-SocketIO/blob/main/src/flask_socketio/__init__.py#L653
            - Call 'run_server()' function
            - Import eventlet

- Then it will get address information from our ‘host’ and ‘port’
- ‘Eventlet’ will listen on our host and port
    - eventlet_socket = eventlet.listen(addresses[0][4], addresses[0][0]) (line 662)
    - Link: https://github.com/miguelgrinberg/Flask-SocketIO/blob/main/src/flask_socketio/__init__.py#L662
    - Call convenience.listen (line 38)
    - Link: https://github.com/eventlet/eventlet/blob/master/eventlet/__init__.py#L38
        - Call listen() function (line 34)
        - Link: https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/convenience.py#L34
        - listen() function will listen on the address that passed into the parameter
        - listen() function will return the listening green socket object
            - Uses socket module
            - Link: https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/green/socket.py
            - Call create_connection() function (line 29)
            - Link: https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/green/socket.py#L29
            - Connect to *address*(a 2-tuple "(host, port)") and return the socket object.
            - In create_connection() function, it will get all informations about the address, and call socket() function

on those informations, and call connect() to create the connection.

- Check if we provided a SSL agrament
- Call ‘server()’ function in eventlet/wsgi.py (line 884)
- Link: https://github.com/eventlet/eventlet/blob/master/eventlet/wsgi.py#L884
    - In ‘server()’ function (line 884)
        - This ‘server()’ function will start up a WSGI server handling request from the supplied server socket. This function will loop forever.
        - ‘Sock’ passed in ‘server()’ function: server socket, must be already bound to a port and listening. We pass “eventlet_socket” in here, we got the “eventlet_socket” from line 662
        - Link: https://github.com/miguelgrinberg/Flask-SocketIO/blob/main/src/flask_socketio/__init__.py#L662
        - ‘Site’ passed in ‘server()’ function: WSGI application function.
        - While ‘is_accepting’ == True; (line 1000)
        - Link: https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L1000
        - (The value of ‘is_accepting’ set to True as default at line 31)
        - Link: https://github.com/eventlet/eventlet/blob/85290a11dd5c5f511a45f4010b512eedfef87000/eventlet/wsgi.py#L31
            - Our sock accept the client_socket and client_address
            - Connect the client_socket and client_address to our sock
- Then call run_server() again (line 682 - 685)
- Link: https://github.com/miguelgrinberg/Flask-SocketIO/blob/main/src/flask_socketio/__init__.py#L682

- This will call run_server over and over again