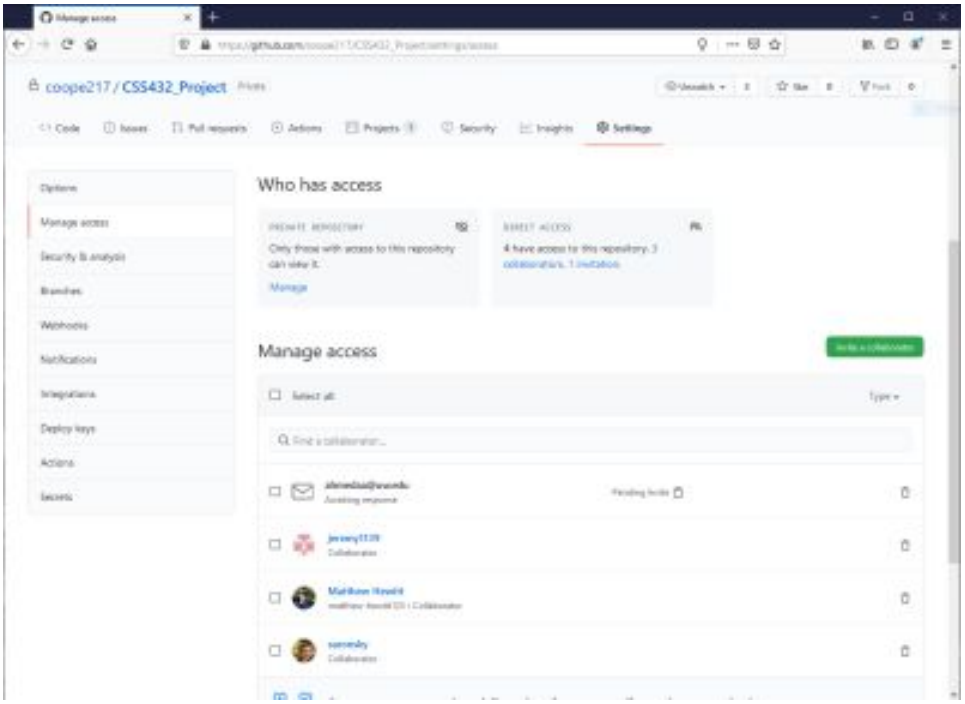
Network Noggins Documentation

By: Matthew, James, Jeremy, and Simon

Github Link (Version Control)

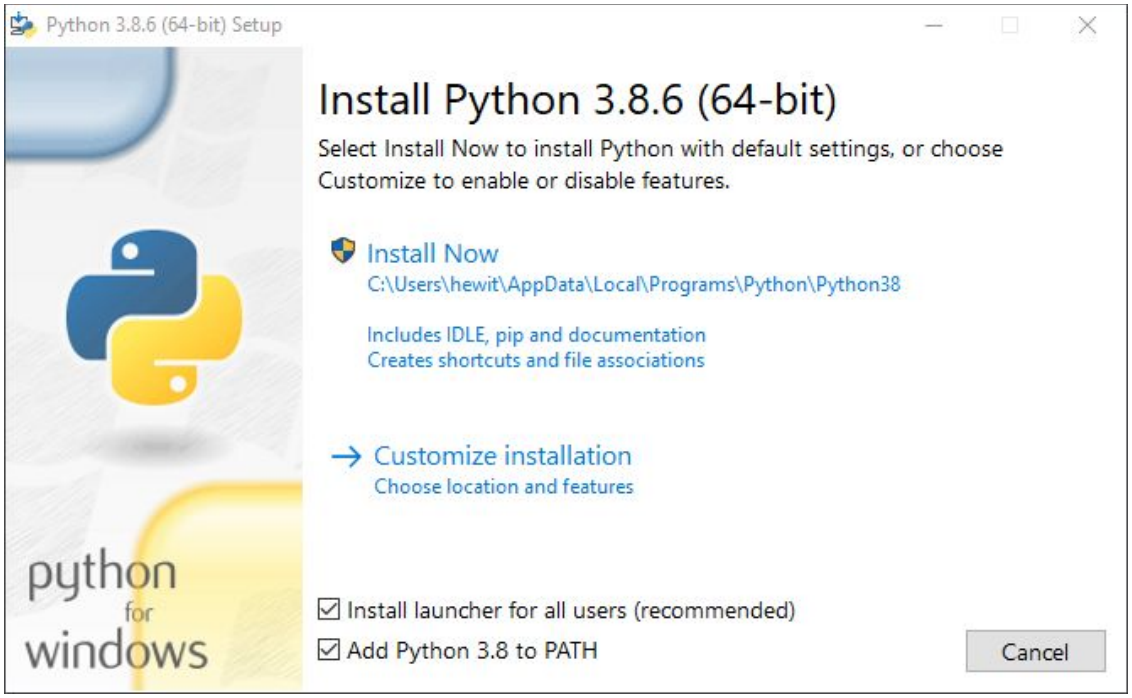


[Github]

Instructions For Running The Code

In order to run the Tic-Tac-Toe project the following preliminary requirements must be met first:

- Python version 3.8.6 is installed on the client-side. Note: You may need to select the option, “Add Python 3.8 to PATH” so that Python works accordingly.



- A connection to UW’s VPN must be active in order to connect to the server.
- *tic_tac_toe_server.py* must be on *csslabX.uwb.edu* (X = CSS lab number where the server is stored)
- *tic_tac_toe_client.py* must be stored locally somewhere on your machine.

Setting Up the Server

Once the requirements are met, connect to *csslabX.uwb.edu* using a terminal (ie PuTTy) and navigate to the directory where *tic_tac_toe_server.py* is stored. When you navigate to the correct directory, type *python tic_tac_toe_server.py* in order to set up the server and allow the server to begin accepting connections from clients. The server should ask you for a port number upon running. An example port number we used for the project was 4001 but most port numbers will do.

Setting Up the Client

Once the requirements are met, download *tic_tac_toe_client.py* in any corresponding directory. Navigate to that directory from the command line and run the client by typing *python tic_tac_toe_client.py* in order to connect to the server and initiate gameplay. The client will ask for server address and port number. Input the address such as *csslabX.uwb.edu* with X being

the number of the machine where the server is running. The port number will be the same port number you used to listen to client connections back in the server side.

Instructions For Playing the Game

Pre-game setup

At the initial startup, clients are introduced with a field where they can type their name. This name is used during games to identify who is currently connected. Once the user enters their name, they will be presented with a lobby list of available players to challenge in a tic-tac-toe game. From here, they can choose to join or create a game (or refresh if there are no available players showing up on the list).

In-Game

Once users enter a lobby or create a game and have someone join it, gameplay will begin and each player can make their moves by entering a numerical value between 1-9 which represents a location on a tic-tac-toe board.

Post-Game

After a game has concluded and a user wins or draws, they will return to the lobby list and be able to challenge a new user. Users can decide to leave the lobby and unregister their name anytime by simply disconnecting or closing the program or typing ‘E’ when they are at the lobby to exit the application.

Protocol Implementation

Register

When users run *tic_tac_toe_client.py* and input server address and port number, they are presented with a window to input their name to send to the server. The server will then store their name in order to display their name in a lobby list and set-up games for the specified users.

List Games

When users enter their username after the initial startup, a lobby list will be presented to them with the names of currently connected players that they can initiate a tic-tac-toe game with.

Create Game

This is implemented alongside the register requirement where once a user inputs their name and presses enter, the server will register the user with their name and automatically place them in a list of available users to initiate a tic-tac-toe game with.

Join Game

This is implemented alongside the list games requirement where the user can use the lobby list in order to identify which user they want to play with and select the desired user within the list to initiate a tic-tac-toe game with them.

Exit Game

Players can unregister by simply terminating the program. Alongside ‘Unregister’, players can choose to exit the game manually by typing ‘E’ into the console when they are at the lobby screen.

Unregister

Alongside ‘Exit Game’, when a player is in the lobby, they can choose to exit and unregister themselves by typing ‘E’ into the console.

Application Specific Protocol

Client

The client will be expected to send three different types of data. Each data type correlates to a state that the client is currently in. The following states are:

- Startup (username data)
    - The client will send a string of letters for the server to store and identify a client’s username.
- Lobby state (command letter or number)
    - The client will send a letter in order to perform different actions or a number to join a created game when sitting on the lobby screen.
- Ingame (move number)
    - The client will send a number in order to make a move on a tic-tac-toe board.

Server

The server will be able to handle each state individually for each user since client connections are divided into threads to handle lobby commands and ingame commands separately.

Bonus Features

Game Video

Video demonstrating how gameplay looks is attached to the submission.

Scoreboard

Players currently on the lobby screen can type ‘S’ to display a scoreboard of every currently connected player’s win/loss record.

Chat Room

A chat room is available while a client is waiting in a lobby.