Tic Tac Toe

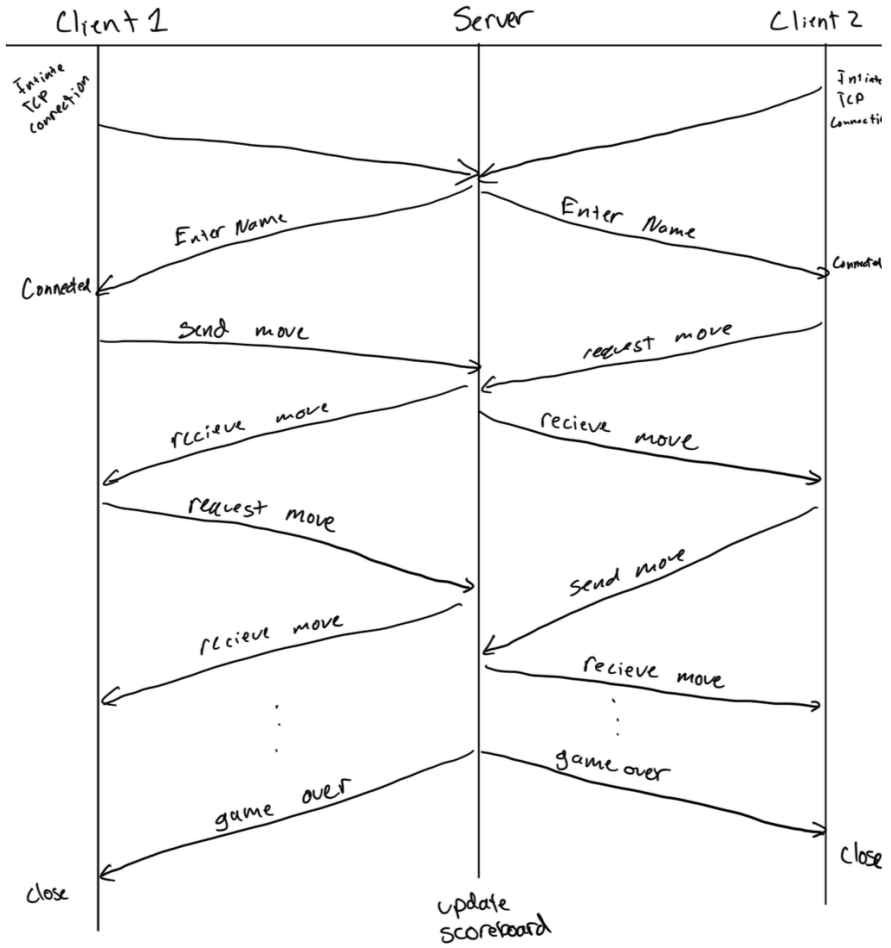
Isai Tinoco Gutierrez

Russell Ferrall

Application Architecture

- Client-Server
- Multithreaded server
- Thread-safe lock scoreboard

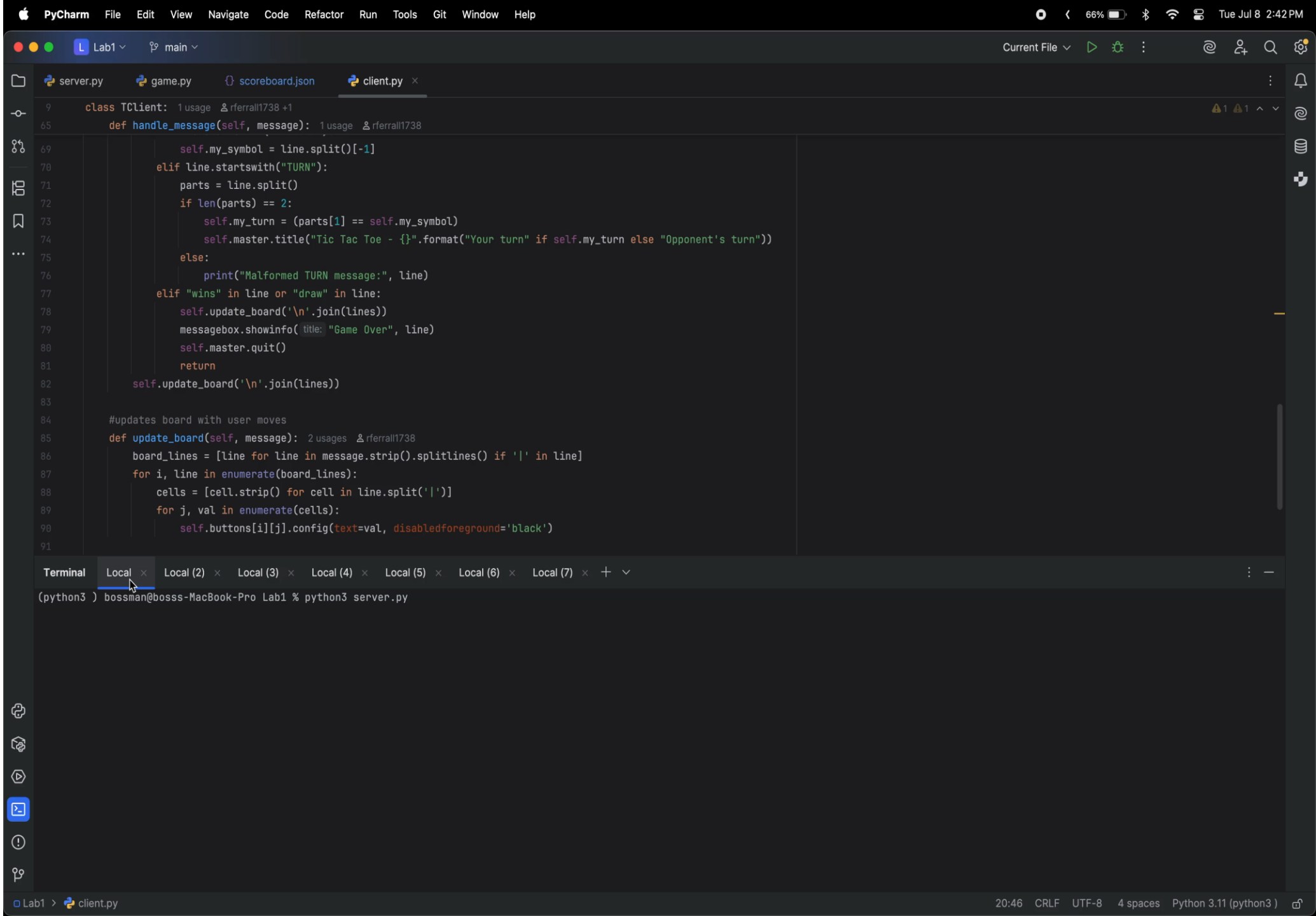
System Components Diagram



TCP

- Used TCP Sockets
- Ensures no dropped or duplicated moves
- Chosen for:
  - Reliable delivery
  - In-Order communication

Demo


PyCharm File Edit View Navigate Code Refactor Run Tools Git Window Help
Tue Jul 8 2:42 PM
Lab1
main
Current File
server.py
game.py
scoreboard.json
client.py
class TClient: 1 usage rferrall1738 +1
def handle_message(self, message): 1 usage rferrall1738
self.my_symbol = line.split()[-1]
elif line.startswith("TURN"):
parts = line.split()
if len(parts) == 2:
self.my_turn = (parts[1] == self.my_symbol)
self.master.title("Tic Tac Toe - {}".format("Your turn" if self.my_turn else "Opponent's turn"))
else:
print("Malformed TURN message:", line)
elif "wins" in line or "draw" in line:
self.update_board('\n'.join(lines))
messagebox.showinfo(title: "Game Over", line)
self.master.quit()
return
self.update_board('\n'.join(lines))
#updates board with user moves
def update_board(self, message): 2 usages rferrall1738
board_lines = [line for line in message.strip().splitlines() if '|' in line]
for i, line in enumerate(board_lines):
cells = [cell.strip() for cell in line.split('|')]
for j, val in enumerate(cells):
self.buttons[i][j].config(text=val, disabledforeground='black')
Terminal
Local
Local (2)
Local (3)
Local (4)
Local (5)
Local (6)
Local (7)
(python3) bossman@bosss-MacBook-Pro Lab1 % python3 server.py
Lab1 > client.py
20:46 CRLF UTF-8 4 spaces Python 3.11 (python3)

Types, Syntax, and semantics for request + response messages

Name Prompt - "Enter your player name:"

- Requested by server; client responds with a string name.
- **self.s.send(self.name.encode())**
- **name = client1.recv(1024).decode().strip**

Moves

- Sent by clients from button click to the server
- **self.s.send(f”{row} {col}”.encode())**
- **move_data = current_client.recv(1024)**
- Server updates the board for both clients
- **c.send(f”\n{board}\n”.encode())**

Types, Syntax, and Semantics

- Client sends name once at startup via GUI input
- Server manages turn order and sends updates only to the correct client
- Clicking a cell sends a `(row, col)` move to server automatically
- Server validates the move, updates the game state, and notifies both clients
- Game outcome is displayed graphically when the game ends

Rules for determining when and how a process sends/responds to messages

- **Server sends first message –** prompts for player names when clients connect.
- **Clients only respond when prompted** either with name or a move.
- **Turn-based interaction –** server tracks turn order and sends prompts to each player on who's turn it is.
- **Board updates broadcasts to both clients** after every valid move.
- **Game ends with result message** to both clients, followed by socket closure and score update.
- If one player wins, they get 2 points. If it's a draw each player gets 1 point.