

Assignment 08: Mini Project- Hangman Game using Client Server Architecture in Distributed System

```
# hangman_server.py
import socket

def handle_client(conn, addr):
    word_to_guess = "nonchalant"
    guessed = ['_' for _ in word_to_guess]
    attempts = 5
    already_guessed_letters = set()

    try:
        conn.send("👋 Welcome to Hangman!\n".encode())

        while attempts > 0 and '_' in guessed:
            conn.send(f"\nWord: {' '.join(guessed)}\nAttempts left: {attempts}\nGuess a letter: ".encode())

            guess_data = conn.recv(1024)
            if not guess_data:
                print(f"Client at {addr} disconnected.")
                break

            guess = guess_data.decode().strip().lower()

            if len(guess) != 1 or not guess.isalpha():
                conn.send("⚠ Invalid input. Enter a single alphabet.\n".encode())
                continue

            if guess in already_guessed_letters:
                conn.send("❌ You've already guessed '{guess}'. Try a different letter.\n".encode())
                continue

            already_guessed_letters.add(guess)

            if guess in word_to_guess:
                match_found = False
                for idx, letter in enumerate(word_to_guess):
                    if letter == guess and guessed[idx] == '_':
                        guessed[idx] = guess
                        match_found = True
                if match_found:
                    conn.send("✅ Correct!\n".encode())
                else:
                    conn.send(f"❌ '{guess}' is already fully revealed.\n".encode())
            attempts -= 1
    except Exception as e:
        print(f"Error: {e}")
    finally:
        conn.close()
```

```
        else:
            attempts -= 1
            conn.send("X Wrong guess.\n".encode())

        if '_' not in guessed:
            conn.send(f"\n🎉 Congratulations! You guessed the word:
{word_to_guess}\n".encode())
        else:
            conn.send(f"\n💀 Game over. The word was:
{word_to_guess}\n".encode())

    except ConnectionResetError:
        print(f"Connection lost with {addr}")
    finally:
        conn.close()

# Start the server
HOST = 'localhost'
PORT = 5555

server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server.bind((HOST, PORT))
server.listen(1)
print(f"🔗 Server started on {HOST}:{PORT}. Waiting for a connection...")

conn, addr = server.accept()
print(f"🔗 Connected by {addr}")
handle_client(conn, addr)
```

```
# hangman_client.py
import socket

HOST = 'localhost'
PORT = 5555

client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

try:
    client.connect((HOST, PORT))

    while True:
        data = client.recv(1024)
        if not data:
            print("X Server closed the connection.")
            break

        message = data.decode()
        print(message)

        if "Guess a letter" in message:
            guess = input("Your guess: ").strip()
            if guess == "":
                guess = " " # handle empty input
            client.send(guess.encode())
        elif "Congratulations" in message or "Game over" in message:
            break

except ConnectionRefusedError:
    print(" ! Could not connect to server. Is it running?")
except Exception as e:
    print(f"⚠ Error: {e}")
finally:
    client.close()
```

The screenshot shows a Windows desktop environment with several open windows. In the center is a code editor window titled 'BRO CODE PY' containing the 'hangman_server.py' and 'hangman_client.py' files. The 'hangman_server.py' file is on the left, and the 'hangman_client.py' file is on the right. Both files are written in Python and handle socket communication for a hangman game.

hangman_server.py

```
PS C:\Users\kamat\Desktop\BRO CODE PY> python hangman_server.py
Server started on localhost:5555. Waiting for a connection...
Connected by ('127.0.0.1', 53426)
PS C:\Users\kamat\Desktop\BRO CODE PY>
```

Word: o _ _ a _ a _ - -
Attempts left: 4
Guess a letter:
Your guess: t
 Correct!

Word: _ o _ _ a _ a _ t
Attempts Left: 4
Guess a letter:
Your guess: l
 Correct!

Word: _ o _ _ a l a n t
Attempts Left: 4
Guess a letter:
Your guess: c
 Correct!

Word: n o n c _ a l a n t
Attempts left: 4
Guess a letter:
Your guess: h
 Correct!

Congratulations! You guessed the word: nonchalant

PS C:\Users\kamat\Desktop\BRO CODE PY>

hangman_client.py

```
PS C:\Users\kamat\Desktop\BRO CODE PY> ...
# hangman_client.py
import socket
HOST = 'localhost'
PORT = 5555
client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
try:
    client.connect((HOST, PORT))
    while True:
        data = client.recv(1024)
        if not data:
            print("X Server closed the connection.")
            break
        message = data.decode()
        print(message)
        if "Guess a letter" in message:
            guess = input("Your guess: ").strip()
            if guess == "":
                guess = " " # handle empty input
            client.send(guess.encode())
            elif "Congratulations" in message or "Game over" in message:
                break
        except ConnectionRefusedError:
            print(" Could not connect to server. Is it running?")
        except Exception as e:
            print("▲ Error: (e)")
finally:
    client.close()
```