Matthew Allpass

9/20/24

CS 457

Statement of Work

Project Title:

CS457 Game Project

Team:

Matthew Allpass

Project Objective:

This goal of this project is to have a Python based quiz show that uses a client-server architecture. Each client (player) will be able to connect to the server where they will be able to answer questions in real time with other players. The server will be responsible for keeping track of the game state as well as handling client connections.

Scope:

Inclusions:

- Sockets & Client-Server architecture
- Game logic & Quiz question bank
- Multiplayer support
- Error Handling
- Help and Documentation
- Command line interface

Exclusions:

- Unless time permits, there will be no GUI
- Single-player mode/Play against an AI
- Persistent data storage

Deliverables:

- Sprint 1: TCP Client and Server
- Sprint 2: Game logic/message protocol & Managing client connections
- Sprint 3: Multiplayer Functionality

- Sprint 4: Game state & play
- Sprint 5: Error handling

Timeline:

Key Milestones:

- Sprint 1: 10/6
- Sprint 2: 10:20
- Sprint 3: 11/3
- Sprint 4: 11/17
- Sprint 5: 12/6

Task Breakdown:

- Sprint 1
    - Implement TCP socket connection between server and client
    - Test basic communication between server and client
- Sprint 2
    - Design protocol to send quiz data
    - Design protocol to send player responses
    - Ensure server can handle multiple client connections simultaneously
    - Build quiz question bank
- Sprint 3
    - Design multiplayer logic
    - Ensure proper synchronization of game state across all clients
- Sprint 4
    - Design game logic (scoring, determine winners, turns)
    - Handle game completion
- Sprint 4
    - Implement error handling for invalid inputs
    - Implement error handling for common network issues

Technical Requirements:

Hardware:

- Server machine capable of running python and handling multiple client connections (CS department machine)
- Machine for client capable of running python and handling basic networking tasks (CS Department machine)

Software:

- Python
    - Socket library, threading, argparse, time
- Any OS (mac, windows, linux) will work but linux is what it will be tested off of

Assumptions:

- Network Connectivity
- Access to CS Department machines
- Basic terminal understanding

Roles and Responsibilities:

- Matthew will do everything

Communication Plan:

- GitHub will be used; goal is to have at least 1 push per week with updated documentation