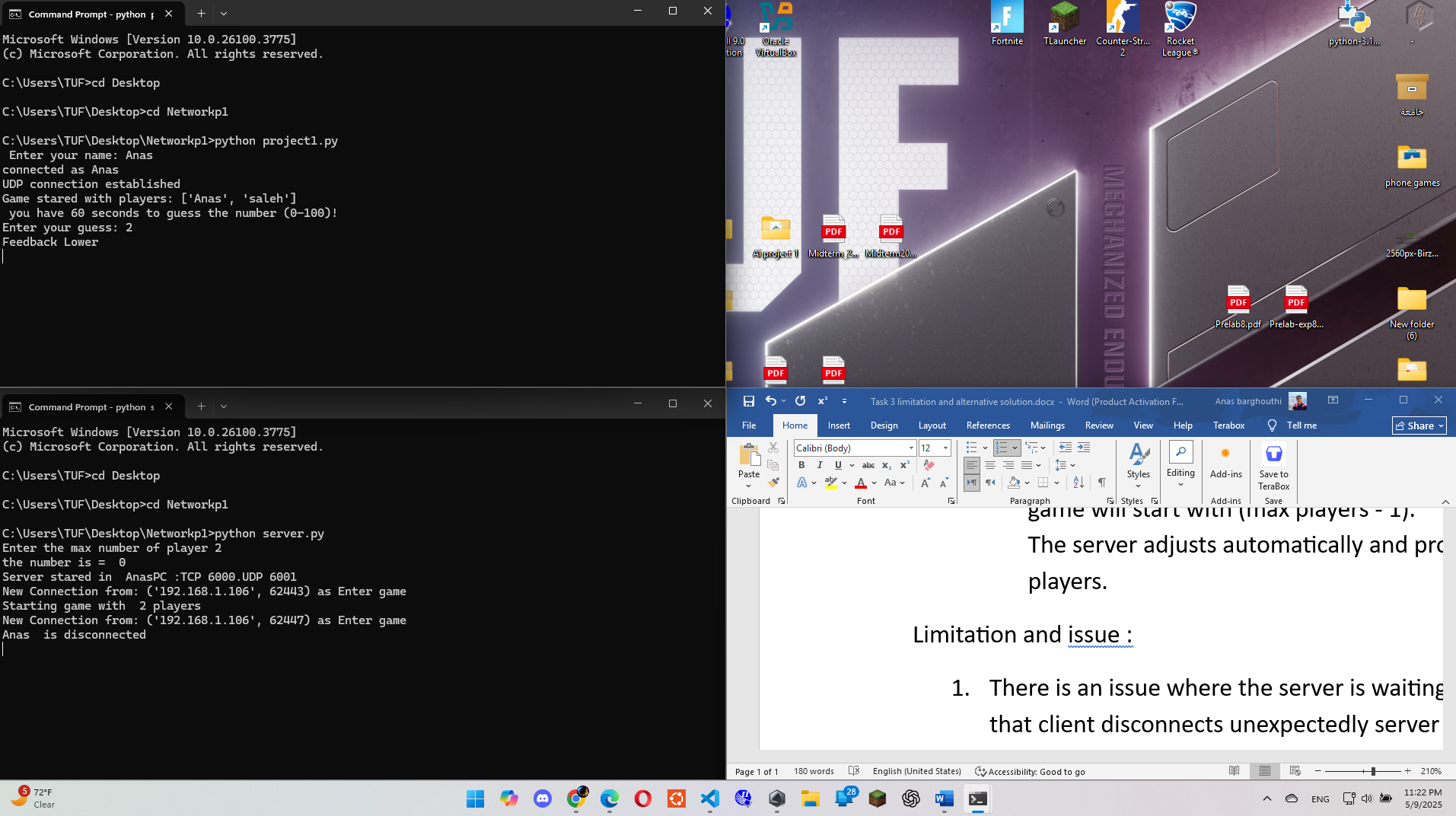
Alternative Solutions:

1. In the project instructions, a fixed maximum number of players is defined for the game.  
   However, in our implementation, the maximum number of players is provided as input when starting the server.  
   This approach allows the server to handle different game sizes dynamically (e.g., 2, 3, 4, 5 players, etc.).  
   It can even handle a session with just one player, if the maximum number is set to 1.
2. If a player connects to the server but disconnects right after entering their name, the game will start with (max players - 1).  
   The server adjusts automatically and proceeds with the remaining connected players.

Limitation and issue :

1. There is an issue where the server is waiting for a client to guess the target number, but that client disconnects unexpectedly server will stuck also another client will enter a problem. Since we are using a UDP socket, which is a connectionless protocol, the server cannot detect whether a client is still connected or not.  
   As a result, when the disconnected client doesn't respond, the server appears stuck.



During the game, Saleh disconnected at the moment he was supposed to make a guess. However, the server mistakenly identified Anas as the one who disconnected. As a result, the server continued listening to Saleh but treated him as if he were Anas. This misidentification caused the server to become stuck, which in turn affected the responsiveness of all other connected clients.