****

CS434 - INTERNETWORKING PROTOCOL

**LAB 01: SOCKET PROGRAMMING**

**Self-Evaluation**

1. **Team information**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Name** | **Student ID** | **Contribution(%)** |
| 1 | Đinh Vũ Quỳnh | 1751099 | 43 |
| 2 | Phan Đăng Khoa | 1751124 | 32 |
| 3 | Phạm Quốc Huy | 1751072 | 25 |

1. **Score sheet**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Requirements** | **Score** | **Evaluate** |
| 1 | Use C/C++, Java,C# | 2 | 2(C#) |
| 2 | Implement wholegameplay properly | 3 | 3 |
| 3 | Socket Non-blocking | 2 | 2 |
| 4 | Have a good UI(MFC,WPF,Swing) | 3 | 3 |
|  | Total | 10 | 10 |

1. **Specifications**

Storyline: The story of the game is how the requirements are stated(i.e all requirements are implemented). *In short*: The server sends some keywords which the clients(players) will guess by switching between players(with some constraints included). At the end, the player who has the most point wins

Bonus functions:

**Scoreboard**: There is a scoreboard ranking players when the game end.

**Status board**: We do this because we think it is intuitive and make the game more easy to play. It shows the status of the game(a new question loaded, announce the score earned, rightness of the character/keyword sent, instructions, status of players,…

**Chat between players and server**: Player can send message between clients

**Source code design**

We design the system as 2-Tier architecture. Presentation layer and Logic layer(Data layer is eliminated because it is too simple to count as one)

There are two main entities in the system

|  |
| --- |
| **Question** |
| keyword: String |
| hint: string |
| size: int // size of keyword |
| showed: List// *the the indices of characters that have been shown* |

|  |
| --- |
| **Player** |
| mickname: string |
| point: int |
| no\_turn: // *the number of time that this user has been guessed* |
| disqualified: Boolean;// *have this player been disqualified or not* |
| socket: Socket |

Logic of sending-receiving: establish threads dedicated to the business of receiving and sending message from server and players(clients).

As clients receive message from server, we include a flag as a capitalized with size of 4 characters in the beginning of the message, base on this, the one who catch these flags will know what the other end is wanting and do the business accordingly.

Here are some flags that we include

“QQQ:”: attach with the question and hint sent from server to players in the first place

“ONE:”: one character sent from the player

“ALL:” the player wants to guess the whole keyword

“MSG:”: chat message

“COR:” tell that the player has guess it right

“FLS:”: wrong guessing sent from server

“ANNO”: general announcement from the server

“UDT:” includes the keyword attached about the new keyword that has been updated

“TURN”: whether to not give the right to play in the turn for one player sent from server