IT328 – Network principles

Second semester 1445

Project description

جامعة الملك سعود
King Saud University
1957

Total Grade	**10%**
Outline of Problem	The common language of the Internet is IP. Programmers who create networked applications typically interface with IP via either the User Datagram Protocol (UDP) or the Transmission Control Protocol (TCP). Sockets offers an Application Programming Interface to both UDP and TCP. In this project you are to investigate the Sockets interface and produce a small networked application as outlined below.
Problem	Online multiplayer games are a part of the Internet where all the players connect to the server which in turn communicates to each player. Every action performed by a particular player is sent to the server which in turn sends that action to the rest of the players connected to the server. The following characteristics constrain the design space of an online gaming system: • Number of players is equal to the number of group members. • Real-time communication: the game happens in real time: one person solve the answer, presses "enter," and the others see the updated score. • Based on typed text. The main mode of communication is via text, • Connected over a network. The players involved in the communication may be in physically remote locations and are connected over the internet. Your task will be to design an online gaming system with the above properties, as well as additional properties that you will incorporate into your design. This system will include a server component that handles the transfer of actions and other data, and a client component with a graphical user interface.
Purpose	The purpose of this project is two: • **First**, you will learn several Java technologies, including networking (to support connectivity over a network), sockets and I/O (to support real-time, text-based communication), and threads (to support two or more people communicating concurrently). State machines may be useful to specify certain aspects of the system's behavior. • **Second**, this project will introduce you to the state-of-the-art for enabling human-computer interaction: graphical user interfaces. You will: • become familiar with Swing, a graphical user interface (GUI) toolkit for Java, that is similar to many other such toolkits;

	- learn important GUI programming concepts, including the notion of a view hierarchy and the model-view-controller design pattern; - use event-based programming and the publish-subscribe pattern; - use object modeling notation to explore and express these structures; - confront user interface design challenges. Throughout the project, you will need to design and implement mutable datatypes, paying particular attention to their specifications and how they interact with one another.
Specifications	Implement an online gaming system in Java with the following functions: 1. **Connect**: seems like a straightforward one, surely we need connect event so that we register the **username** in the system and tell him that he is connected once he presses the connect button. Also, presents the names of connected players**.** 2. **Pair Request:** this event signals that a client wants to play a new game (press button Play), so whenever it's triggered, the server should start adding this player to the available room (consider the max number of players) with other players waiting for more people to join. 3. **Player Joined:** once you find a room that needs the player who fired the **pair request,** we should add him to this **waiting list** and notify other players through this event that a new member has just joined the room (list all member in the same waiting room). 4. **Game started:** The server should fire this event to players when either max players are available already or after 30 seconds (with more than one player). The player can see the scores for all players in the same game. 5. **Player Leave request/disconnect** from client side we need a way to get notified as a server that someone left the network -built in disconnect event- or that someone clicked leave and wants to quit the game now, so that we announce to everyone that he left and update the player list. Complete the game unless there is only one player. 6. **Game Ended:** This event should be broadcast-ed from the winner client to the server and from the server to all other clients. If the timer is ended and no winner, end the game.
Phases	There are two phases for this project: For the **First phase (April 15th)**, you will be evaluated during your lab session on: - Functions (1,2,3,4) in previous section (including the interface code that is related to these functions.

	For the **Second Phase (May 1st):** • Functions (5,6) in previous section (including the interface code that is related to these functions).
Resources Required	Java compiler
Hints	• This is group coursework, to be carried out by groups of 3 students. • You must be able to run your gaming system by having one computer as client per each group member (for example: if you have three group members→ you should have four different PCs, a server and three clients). • Each group member must understand every detail in the project and be able to answer any question regarding the evaluation. • Cheating or copying others' solutions is not tolerated. If two solutions are similar, or any other form of cheating has been used, zero grades will be given to both teams.
Game idea	There are two ideas for the game and all related to wording games: **First game:** The server gives the player a set of letters, the player should create words that are composed of the shown letters. (The winner is the fastest one who answers 5 words correctly). For example: The server presents these letters to all players (Adrsowlmenbrit) → The players' answers should be like these: art, bit, draw, lemon, slow. **Second game:** The server gives the players unordered words. The players should send the word in the correct order. (The winner is the fastest one who answers 5 questions correctly). For example: The server presents unordered words to all players and the player who solves 5 questions is the winner and the game is ended. Pplea → apple Lmfi → film Bleat → table
Grading	**Phase 1: 5.5 out of 10** Individual Evaluation and understanding **(1)** Server can maintain an unlimited number of open connections. **(1)** Connection code **(0.5)** Show connected client names **(0.5)** Show the players' names that are waiting in the waiting room **(0.5).** Start the game at the same time for all players. **(0.5)** Updated scores for each player is shown in other players' pages. **(0.5)**

	Complete interface for all functions **(1)** **Phase 2: 4.5 out of 10** Individual Evaluation, understanding **(0.5)** The player can leave the playing room. **(0.5)** If client quits, his name removed form connected list. **(0.5)** Show the winner list after the game**. (0.5)** If no winner, show a message clarify this. **(0.5)** Complete interface for all functions. **(1)** Communication skills **(1)**