Presentation Navigation

a. Problem motivation.

People need to have fun gambling without losing actual money online. An interactive Texas Hold’em

game can be created to satisfy this need.

b. Project goals:

Our project will create an interactive poker game written in Java that can support up to 8 players by the

end of the semester.

**c. Design outline: outline your design choices and decisions and quickly justify each design**

**decision you made.**

a. Design Decisions:

We’ll be following the client-server model. The major components will be the server (game

host), clients (players and spectators), and GUI (one per client).

Server (game host):

One server instance will be created per game. The purpose of the server is to keep track of the

poker game state (e.g. chip amounts and cards for each player, the cards on the table, the pot

and side pots, player actions). It will poll player clients in order and receive their responses and

update the player state according to the user action it received.

Clients (players and spectators):

The client will run on each player’s or spectator’s systems and will send and receive messages

from the server to update its local copy of the game state, which will include just enough

information for the GUI of that player or spectator (spectators will have all players’ cards face up,

while players will only have their cards face up, except in the case of showdown).

GUI (one per client):

The GUI will run on each player’s or spectator’s system in its own thread. The GUI will be

responsible for displaying the onscreen buttons and the actions of players through simple

animations and accompanying sounds. The player’s button presses will be detected and the

corresponding action will be carried out.

**d. Demo - Part I: Demonstrate the main functionality of your project.**

e. Demo – Part II

1. Integrate the already-existing OngoingMode animations to correctly respond to received

GameState from the host to show each player’s actions.

2. Expand the GameState class to include each player’s most recent action so that the

OngoingMode GUI can accurately display status labels for each player.

3. Show Timer and fold

4. Show Quit button

5. Show disconnection (quiting or network error) , update the GameSystem and remove player

**6. Debug the Rank class through adding test cases that handle all possibilities of poker-hands in the**

**TestRank class. (User story 4)**

**7. Debug the GUI portion of the game by play-testing the game with all possible user inputs.**

**(User story 4)**

**8. Debug the GameSystem portion of the game by play-testing the game with all possible user**

**inputs.**

9. Debug the Network (ClientMessageHandler, HostMessageHandler, HostBroadcaster,

HostSearcher) portion of the game by simulating network issues and disconnects and host

Crashing.

f. Good preparation: You speak clearly, your presentation is well organized, your time is

well planned, you don’t exceed your time limit, etc.