

Blackjack Project

By: Tse, Thomas, & Aaron

Motivation

A fun, yet competitive luck-influenced strategic game to play with friends. It is very easy to understand, but has a shockingly high skill ceiling. It will be a good form of digital interaction with lighthearted fun. There will be no actual gambling functionality, so it will serve as a way to play blackjack without monetary risks.

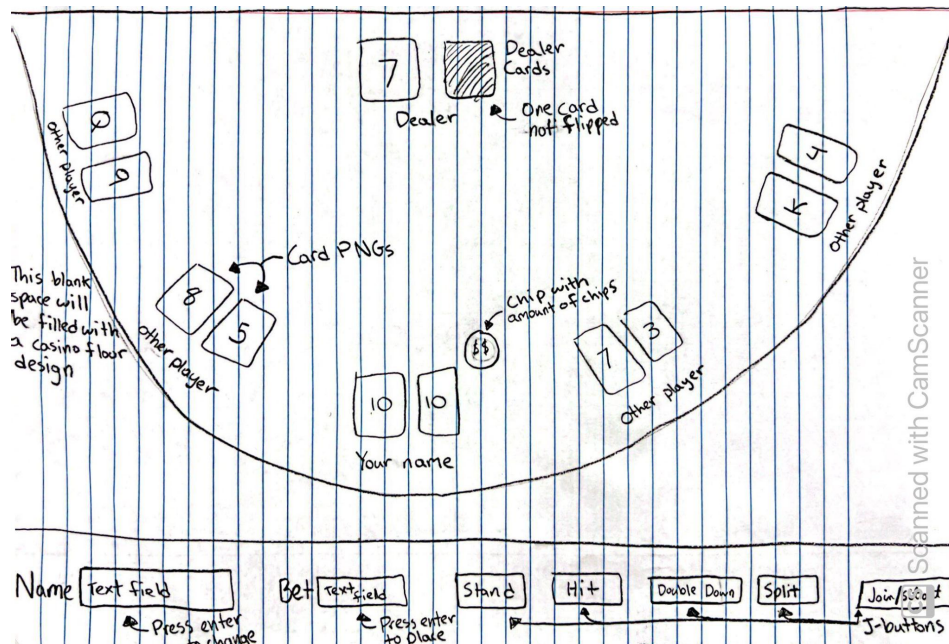
Problem

The main problems we have with creating this game:

1. How will networking be implemented properly to connect users?
2. How can we have the server constantly monitoring the state of the game?
3. What should be the range of chips a user may have? What rules should be in place? Round limits? Different ways to design the game.
4. Frontend design: How do we have it differ for each user without affecting the gameplay/design too much?

GUI

Our primary goal is to have a semicircular table similar to an actual casino, however, figuring out the correct angles to place everything at may be difficult. If this is unachievable, a square table will work just fine with some changes to the GUI. There will be a text field for the client's name, a text field for their betting amount, and buttons to join/sit out a round, stand, hit, double down, and split.



Networking

This will be a multiplayer game that requires networking to send and receive data to and from the server. This data consists of bets, and player moves (hits, stands, double downs, etc.). We plan to implement multithreading to monitor the inputs and outputs from the client and server.

Challenge

Since we are trying to make this as accurate as possible, we will be using 4 decks of cards. This means that we will have to manage 208 cards rather than 52, and make sure that every card has 3 other duplicates. Similarly, you need to continuously update the server output based on client input, as any plays made must affect other players. We will need to allow the game to run with anywhere from 1-5 players, and the experiences should be practically identical.

Testing Plan

We need to thoroughly test a multitude of scenarios, but the initial test will be confirming that 5 clients are able to connect to the server and the sixth is refused

due to the game room being full. We also need to check that the betting amounts are within the bounds that we set (positive, less than x). We will create tests for each of the client actions as well as dealer behavior (must hit until 17, must stand at 17+). The deliverables would include the server that the clients connect to, the threads and sockets used to run each function, and the GUI of the blackjack table because that is how the client will interact with the server.

Manager Workload

Since the manager will be responsible for meetings as well as submissions, the other team members will take responsibility for all of the monotonous tasks such as uploading the card pngs and the GUI organization. All 3 members will work together on the games functionality and error testing, but the manager will not have to worry about the visual aspects of the game.