

Tech Stack:

- Languages: Javascript, CSS, HTML
- Frameworks/Libraries: [React.JS/Node.JS](#)
- IDE: We will all be using our own ide-s, and share our code to Github
- Database: No need for a database, as user information is not stored after they disconnect, so all memory can be handled locally.

Updated plan: Not much has changed in our plan, as the base functionality will be the same. The main difference is that our project is now split into three parts instead of four. As we have begun work on connecting the frontend to the backend we have found that it is best to connect them all through a server client network instead of attempting to connect them locally first. This means that the connection and server will be done together. Outside of that no changes to the original plan have been made.

Update to backend: The backend had to be turned into an API, so this week that was the change that was made. The coded logic was converted into an API so it can be more easily used when talking with the User Interface.

Connections will be made by using JS Websockets and the HTML protocol. While the main, game, and poker modules will be running on the server, the lobby and in game modules will be sent to the browser and run there.

Project Goals:

- Plan A: Create a fully functional Texas Hold 'Em style poker game where multiple users can connect to a poker game and play for fake money. This project should include all of the rules of poker, and a fully functional UI. (we are still on plan A)
- Plan B: If we are struggling to complete this project we will settle for a local poker game, without worrying about server connections.
- Plan A+: If everything is going super well, we will look to add more variations of poker like Pot Limit Omaha or five card stud. (not possible anymore with limited time)

Description of play (to be translated to a computer):

- Each player is dealt two cards from a standard 52 card deck, face down. A round of betting occurs (where each player has the option to check, bet, or fold. Unless a previous player has bet, in which the player only has the options to call (match the bet), fold, or raise until all the players have either checked or put the same amount of chips into the middle)

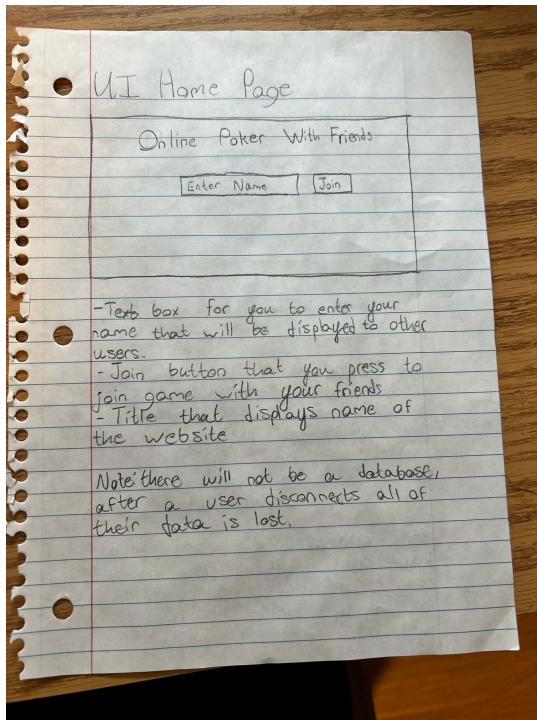
- After the first round of betting, three cards are placed face up in the middle. These cards are called the community cards, and anyone can use them to create the best five card hand. Hand rankings are listed [here](#). Another round of betting occurs.
- After the second round of betting, one more card is added face up to the community cards. Another round of betting occurs.
- After the third round of betting, one final card is added face up to the community cards. A final round of betting occurs.
- If at any point all players except for one “fold” the hand is over and the pot is awarded to the player who did not fold. Otherwise, go to “showdown”
- Showdown occurs when there is more than one player left at the end of the last round of betting. Each player must show their cards, and whichever player has the best hand from the rankings established earlier, that player wins the pot. Repeat the entire game until all players either run out of chips or decide to stop playing.

Description of how a user starts a game and invites friends, or joins a game

- Right now we are looking at creating a web based game, and we are going to experiment between hosting on a server and hosting on one of the player's PCs.
- On the home/start screen of the website there would be an option to either create/host a game or join a friend's game. If the user chooses to create/host a game, the website will use that person's computer to start a connection to host that specific game room. We think this will be more realistic than getting a server set up and always running waiting for somebody to connect to it.
- Once the connection port is established on the host computer we will generate either a shareable HTML link or just a code of 4-6 characters which can be shared with friends so they can connect to your game
- If we use a link there wouldn't be a button for join, the user would just copy and paste it into their web browser and it would take them straight in. If we end up going with the code, then the join screen would just prompt for a code and connect into the corresponding game.

Description of UI:

- Home Screen:
- Note: On this drawing there is only a join button, but there will also be a host a game button that will allow you to enter a room that you yourself created. There also will be a box for you to enter a join code, and you would then enter the code of the room you would like to join if you simply want to join a room instead of creating one. This also means there would be a text box for you to enter the code after you click join game.



- Game screen
- Note: for box 7 it says that the first player to join would have the option to click start game, but this is not specific enough the host of the room (the first person to join) has the option to start the game.

