Team: Nels Anderson
Ehsan Karimi
Justin Visher
Cary Sullivan

Title: OO Poker

Project Summary: A modified Poker Game, specifically Texas Hold ‘em, where users are only given three opportunities to check/fold or bet chips (when dealt cards, after “flop” and after “flip”). We will store player data such as username, winnings, etc. in a database incorporated within our final system.

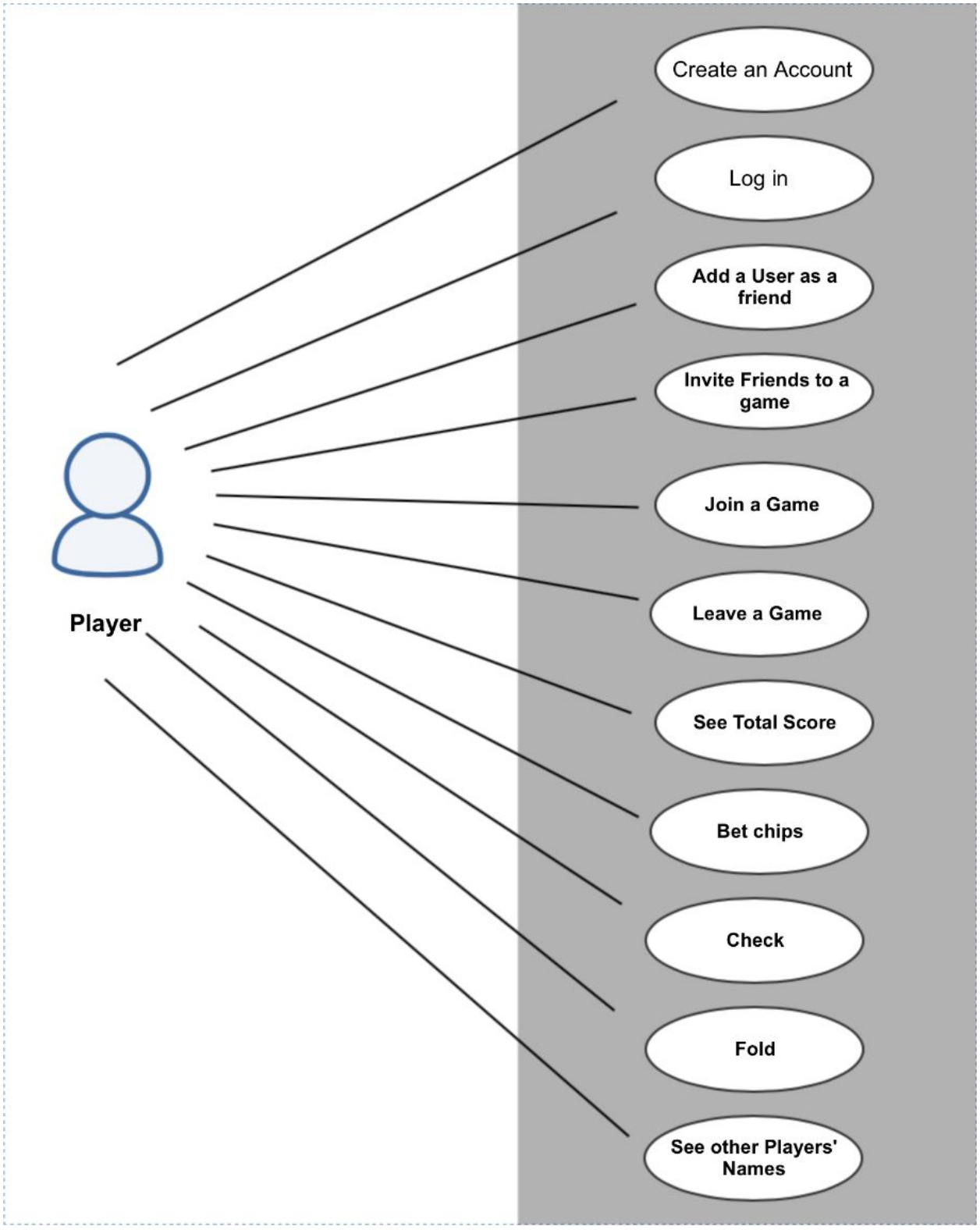
Project Requirements:
There are no Business requirements.

User Requirements		
ID	**Description**	**Priority**
UR-01	As a player, I want to able to create an account so I can play a game.	Critical
UR-02	As a player, I want to log into my existing account so that I can continue my progress over time.	Critical
UR-03	As a player, I want to be able to add other users as friends so I can easily find their accounts again	Medium
UR-04	As a player, I want to invite friends to a game so that we can play together.	Medium
UR-05	As a player, I want to join games so that I can play without starting my own.	Critical
UR-06	As a player, I want to leave games whenever I want so that I don’t have to keep the program running.	High
UR-07	As a player, I want my score kept track of so I can see my progress.	Medium
UR-08	As a player, I want to receive more chips after an hour of running out so that I can continue to play.	Low
UR-09	As a player, I want to be able to bet during a game to increase the current stakes at the table.	Critical

UR-10	As a player, I want to be able to check during a game to avoid spending more chips.	Critical
UR-11	As a player, I want to be able to fold during my turn so that I can end my hand.	Critical
UR-12	As a player, I want to be able to see other player's names so I can see who I am playing with.	Medium

Non-Functional Requirements	
ID	**Description**
NFR-01	Usability - The player must have a general understanding of Texas Hold'em.
NFR-02	Usability - The player must have a general understanding of how to use a computer (clicking links, signing up for a user account, etc.)
NFR-03	Reliability - The system must exit gently in the case of failure.
NFR-04	Reliability - The system should be able to support at least 100 different games of up to 8 players in size
NFR-05	Reliability - The system should store credentials securely, not in plain text.
NFR-06	Reliability - The system must be accessible by player 24/7.
NFR-07	Reliability - The system must automatically save high scores after each score update.
NFR-08	Performance - The system will be able to store up to 500,000 users and scores
NFR-09	Performance - The system must update the game immediately after each interaction by the player.
NFR-10	Supportability - The system should be maintained on a daily basis by the maintenance group.
NFR-11	Supportability - The system must be compatible with common operating systems offered by the market..
NFR-12	Implementation - The player must have access to the internet from his/her platform to get complete access to system features.
NFR-13	Interface - The data must be exported/imported to a server maintained by the maintenance group
NFR-14	Packaging - The system must be installed by the player
NFR-15	Legal- The system must be licensed to the implementation team.
NFR-16	Legal - The development team is not legally responsible for reliability issues with the system

Use Case Diagram

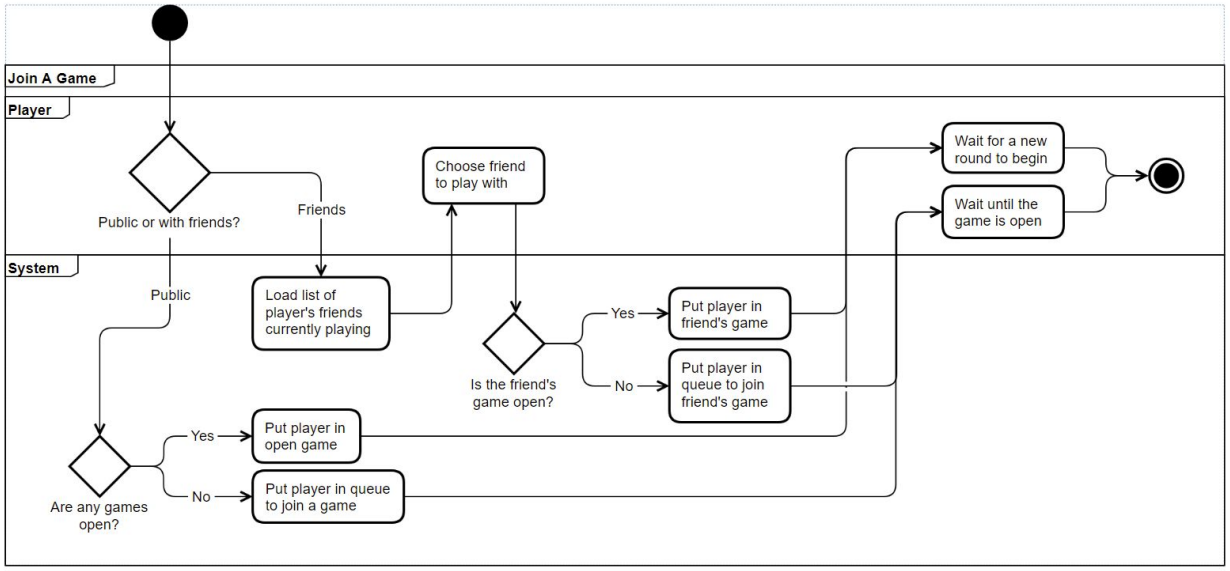


Use Case Documents:

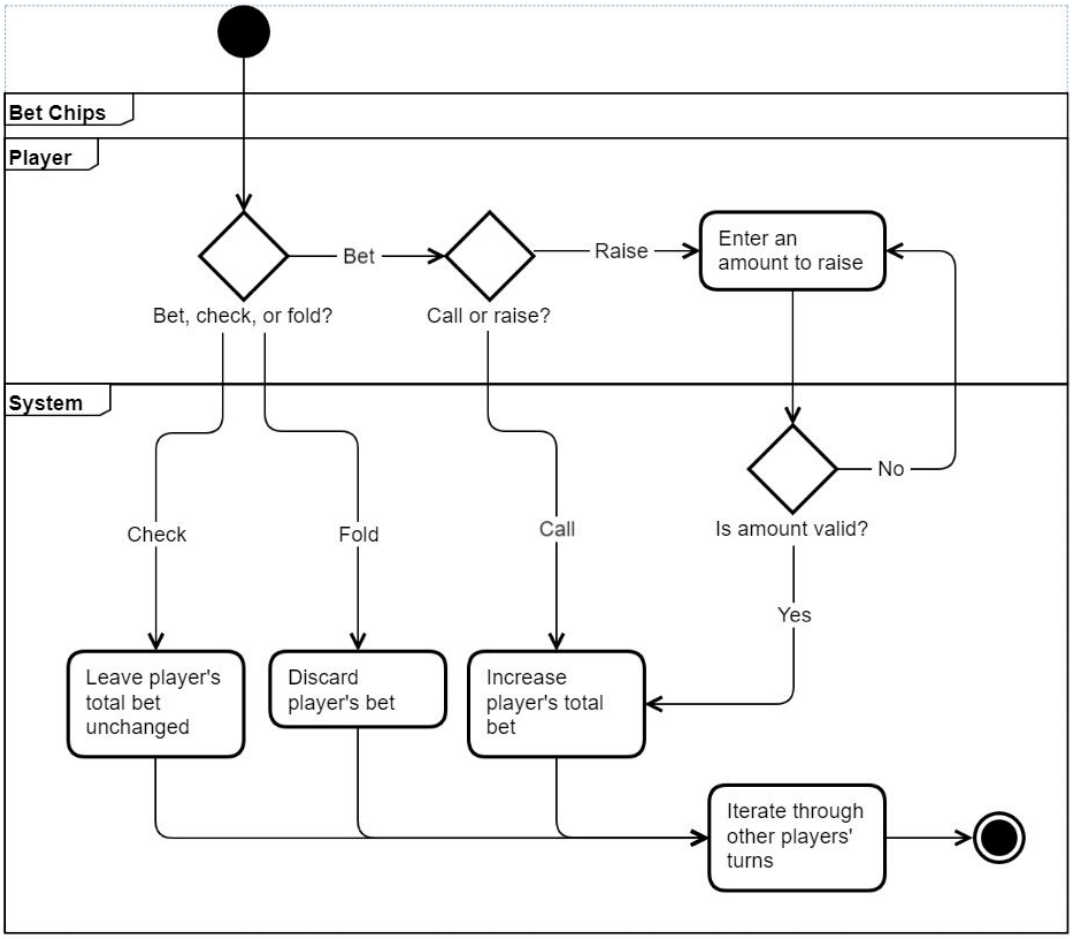
<table>
<tr><td>Use Case ID:</td><td>UC-05</td></tr>
<tr><td>Use Case Name:</td><td>Join a Game</td></tr>
<tr><td>Description:</td><td>Player joins a table to play cards.</td></tr>
<tr><td>Actors:</td><td>Player.</td></tr>
<tr><td>Pre-conditions:</td><td>Player has an account and is signed in.
Player does not start own game.</td></tr>
<tr><td>Post-Conditions:</td><td>Player has spot at table.</td></tr>
<tr><td>Frequency of Use</td><td>Three or more time per week.</td></tr>
<tr><td>Flow of Events:</td><td>
<table>
<tr><th></th><th>Actor Action</th><th>System Response</th></tr>
<tr><td>1</td><td>User selects play game option.</td><td>System prompts user to join a public game or invite friends.</td></tr>
<tr><td>2</td><td>User selects join public.</td><td>System finds game with less than 8 people already in session via joinGame().</td></tr>
<tr><td>3</td><td>User waits until a new round starts.</td><td></td></tr>
</table>
</td></tr>
<tr><td>Variations:</td><td>Player is unable to join a game due to no available spots.
Player joins a game with friends.</td></tr>
<tr><td>Exceptions:</td><td></td></tr>
<tr><td>Developer Notes:</td><td>If all tables are full then a new table will begin.</td></tr>
</table>

<table>
<tr><td>Use Case ID:</td><td>UC-08</td></tr>
<tr><td>Use Case Name:</td><td>Bet Chips</td></tr>
<tr><td>Description:</td><td>Player selects amount of chips to bet - amount left is updated.</td></tr>
<tr><td>Actors:</td><td>Player.</td></tr>
<tr><td>Pre-conditions:</td><td>Player is in a game.
Player has been dealt cards.
Player is either 1. Before “Flop” 2. After “Flop” before “Flip” 3. After “Flip”.</td></tr>
<tr><td>Post-Conditions:</td><td>Player selects chip amount to bet - amount remaining is updated. Chips selected are entered into the table’s pot.</td></tr>
<tr><td>Frequency of Use</td><td>Between 0 and 3 times per game.</td></tr>
<tr><td>Flow of Events:</td><td>
<table>
<tr><th></th><th>Actor Action</th><th>System Response</th></tr>
<tr><td>1</td><td>Player selects bet during their turn.</td><td>System calls bet() method in Choice class</td></tr>
<tr><td>2</td><td>Player enters an amount to bet.</td><td>Amount is accepted by bet(), which calls updatePot() in the Poker class, and updates the player’s current chip count</td></tr>
</table>
</td></tr>
<tr><td>Variations:</td><td>Player bets an amount greater than their current holdings.
Player checks.
Player folds.</td></tr>
<tr><td>Exceptions:</td><td></td></tr>
<tr><td>Developer Notes:</td><td>A conditional to check that player bets an amount equal to or greater than holdings.</td></tr>
</table>

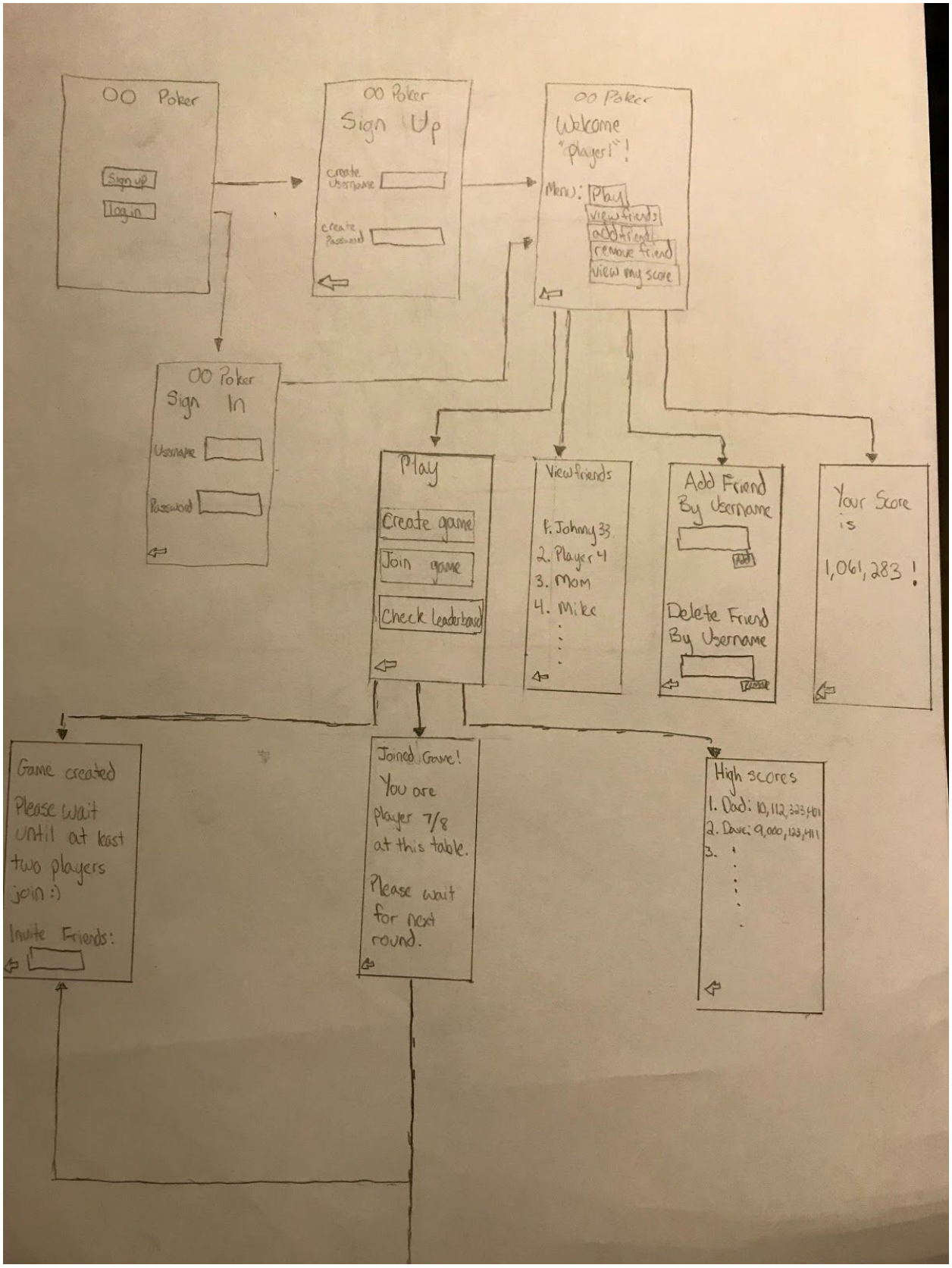
Activity Diagram (UC-05)

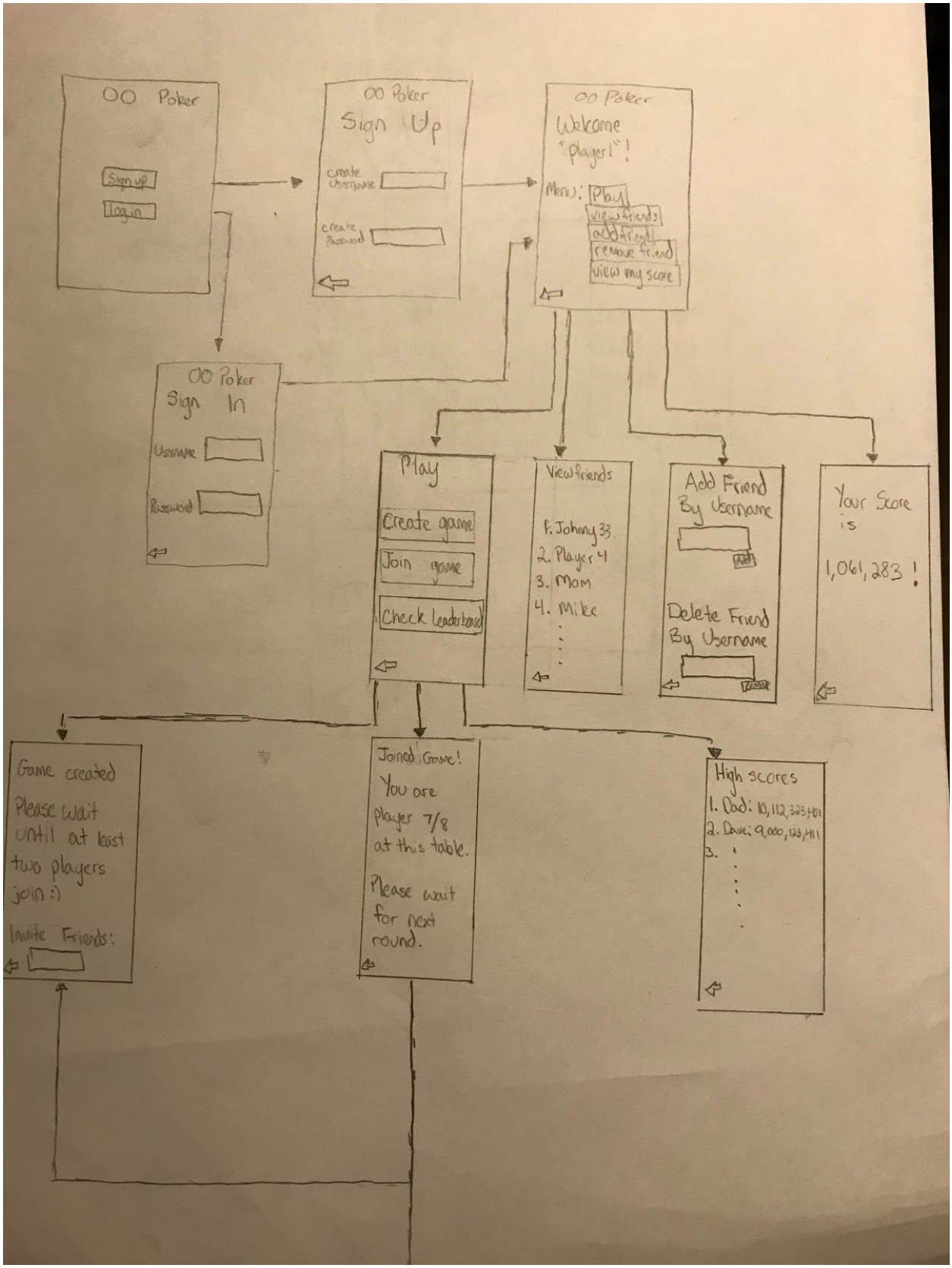


Activity Diagram (UC-08)

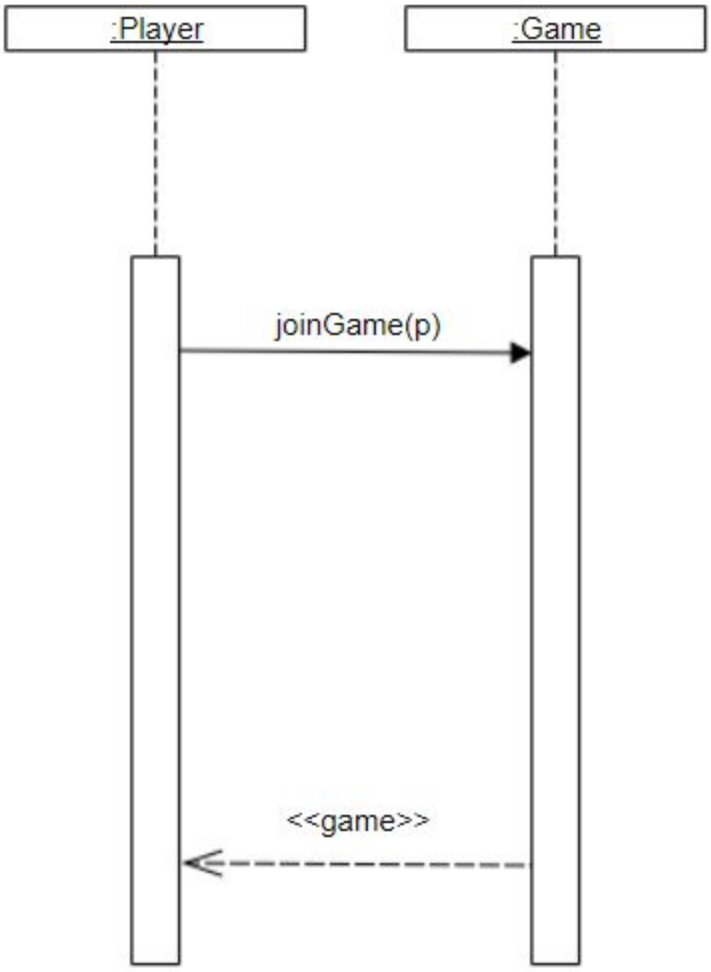


UI Mock-Ups

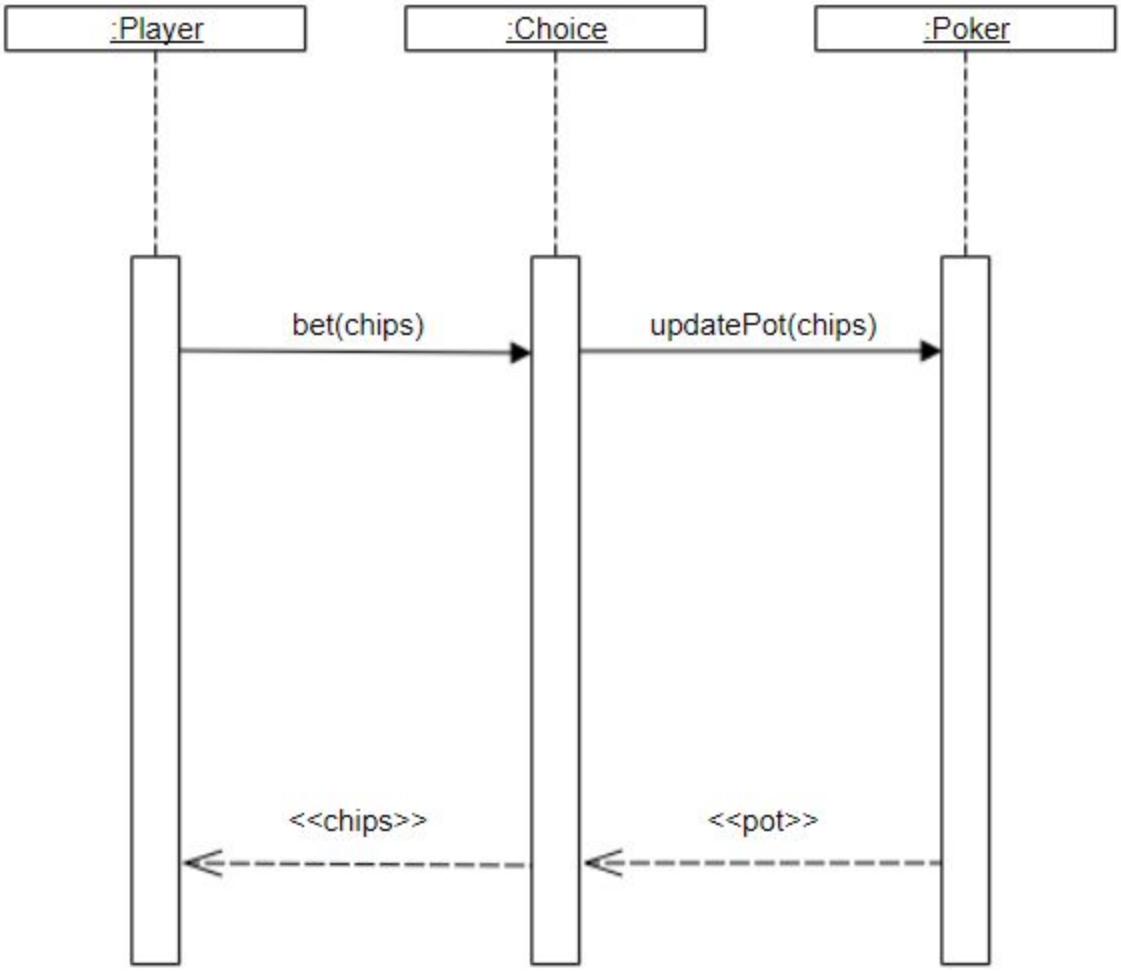

OO Poker
Sign up
Login
OO Poker
Sign Up
create Username
create Password
OO Poker
Welcome "player!"!
Menu: Play
view friends
add friend
remove friend
view my score
OO Poker
Sign In
Username
Password
Play
Create game
Join game
Check Leaderboard
View friends
1. Johnny33
2. Player 4
3. Mom
4. Mike
Add Friend By Username
Add
Delete Friend By Username
Delete
Your Score is
1,061,283!
Game created
Please wait until at least two players join :)
Invite Friends:
Joined Game!
You are player 7/8 at this table.
Please wait for next round.
High scores
1. Dad: 10,112,333,401
2. Dave: 9,000,123,111
3.


OO Poker
Sign up
login
OO Poker
Sign Up
create Username
create Password
OO Poker
Welcome "player!"!
Menu:
Play
view friends
add friend
remove friend
view my score
OO Poker
Sign In
Username
Password
Play
Create game
Join game
Check Leaderboard
View friends
1. Johnny33
2. Player 4
3. Mom
4. Mike
Add Friend By Username
Add
Delete Friend By Username
Delete
Your Score is
1,061,283 !
Game created
Please wait until at least two players join :)
Invite Friends:
Joined Game!
You are player 7/8 at this table.
Please wait for next round.
High scores
1. Dad: 10,112,223,401
2. Dave: 9,000,123,111
3.

Sequence Diagram (UC-05)



Sequence Diagram (UC-08)



Class Diagram