Texas Hold’em Poker Game

CS.4398.252.Group 3

Andrew Whitfield

Felipe Tlusty

Jose F Reyes San Miguel

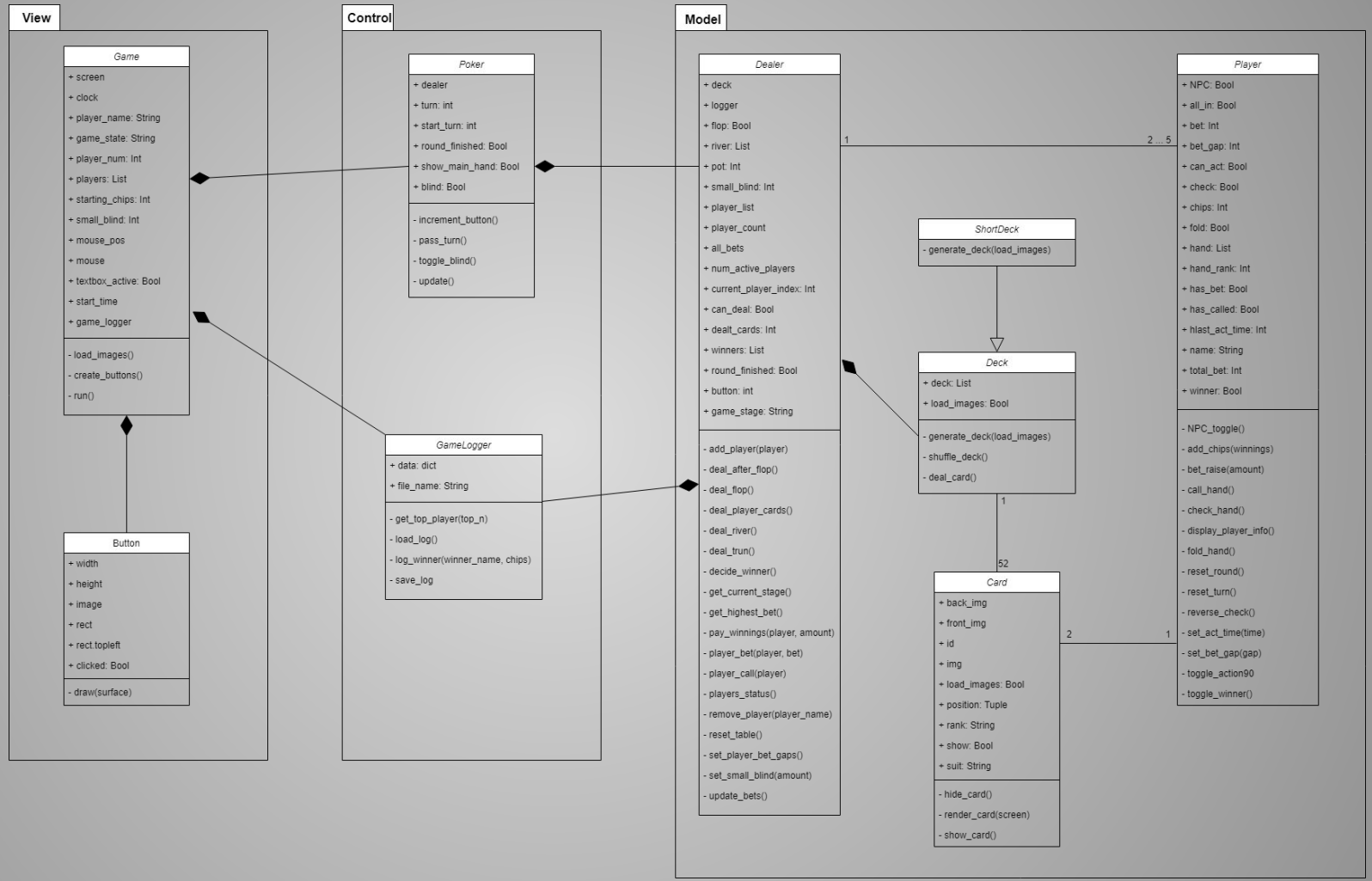
Parker Tindell

Purpose of our system:

Use available tools in python and various libraries such as pygame, as well as some of our own created poker AI, to create a fun simulation of a Texas Hold'em poker game.

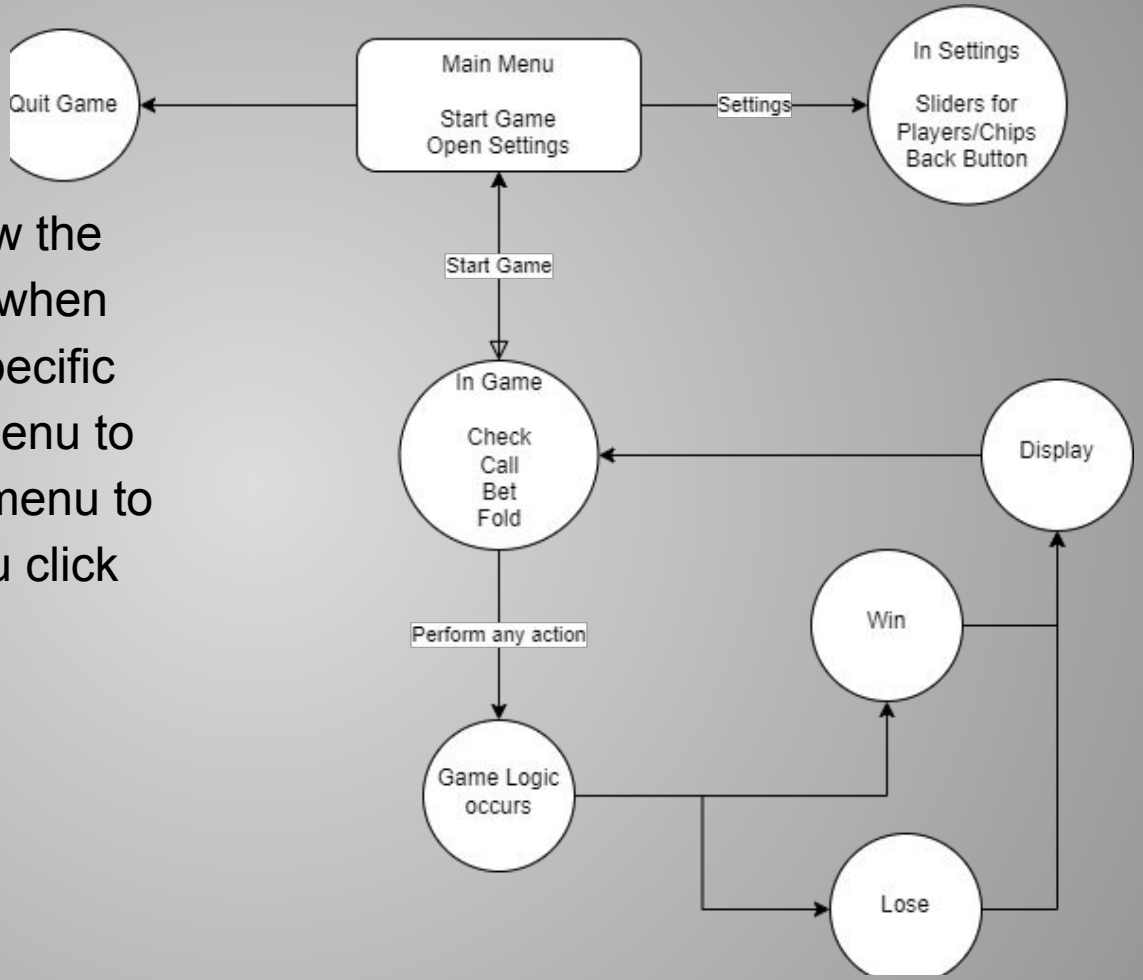
By doing so we walked through the process of software engineering as a group, made statecharts and class diagrams, made acceptance cases and unit tests for our program, and used GitHub to manage our files as we worked on the project.

Class Diagram



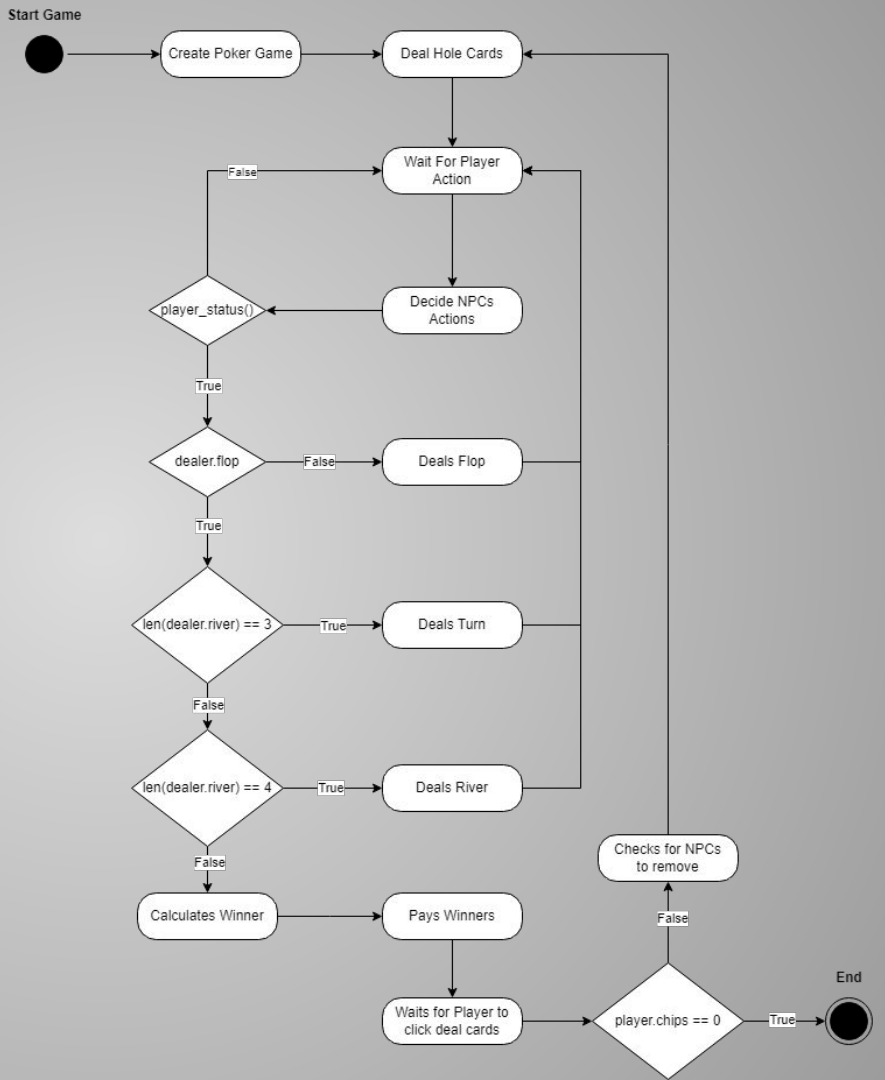
System Statecharts

This state diagram shows how the system goes through menus when the user clicks that menu's specific button. It can go from main menu to settings and back, and from menu to game and back. But once you click quit the game closes.



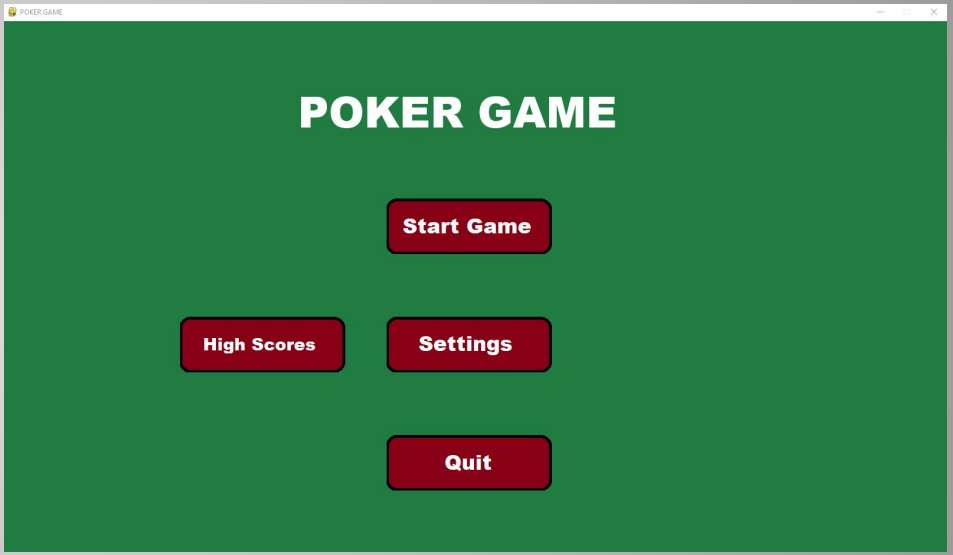
In Game State chart

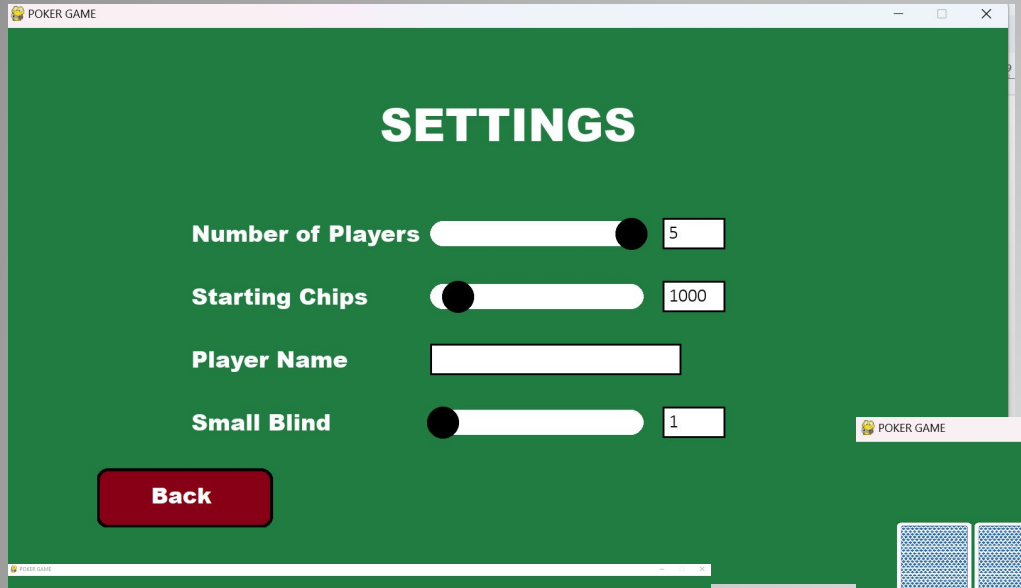
This state chart shows the game logic going through a game round in Poker.
Player actions include things as such: Checking, Calling, Betting, and Folding.

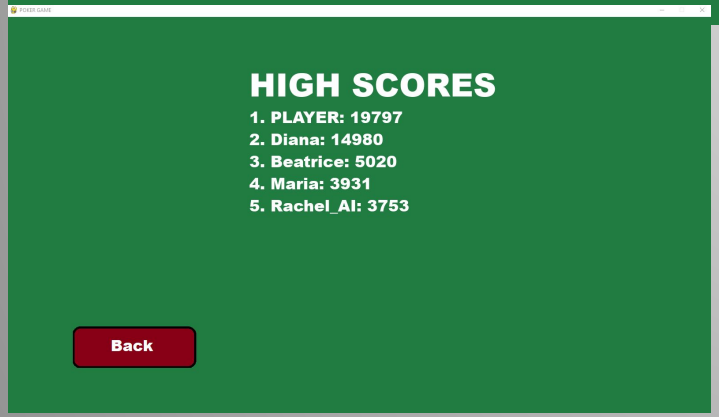


Acceptance Test Cases

1. Game Initialization:
   - Verify that the game initializes without errors.
   - Check that the main menu is displayed upon initialization.
   - Ensure that the game settings are loaded correctly.
2. Menu Interaction:
   - Click on the "Start" button and verify that the game transitions to the in-game state.
   - Click on the "Settings" button and verify that the game transitions to the settings state.
   - Click on the "High Scores" button and verify that the game transitions to the high scores state.
   - Click on the "Quit" button and verify that the game exits without errors.




POKER GAME
SETTINGS
Number of Players
5
Starting Chips
1000
Player Name
Small Blind
1
Back

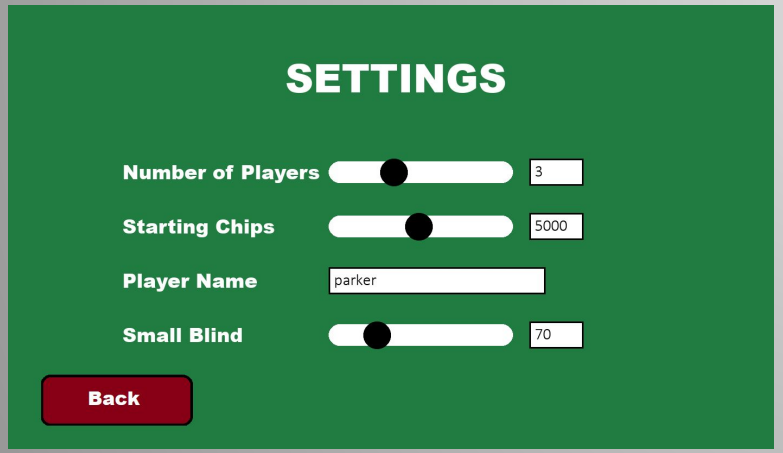

HIGH SCORES
1. PLAYER: 19797
2. Diana: 14980
3. Beatrice: 5020
4. Maria: 3931
5. Rachel_AI: 3753
Back

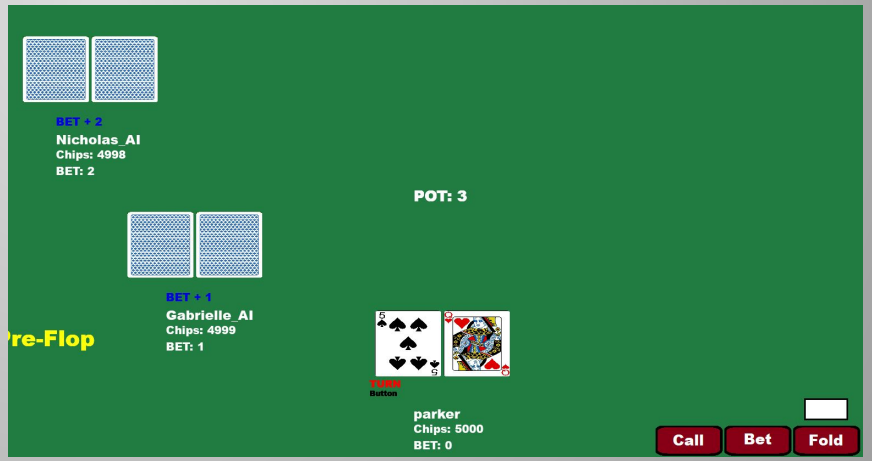

POKER GAME
Leonard_AI
Chips: 1000
BET: 0
BET + 1
Nicholas_AI
Chips: 999
BET: 1
POT: 3
Pre-Flop
Felipe_AI
Chips: 1000
BET: 0
BET + 2
Isabelle_AI
Chips: 998
BET: 2
TURN
Button
Chips: 1000
BET: 0
Call
Bet
Fold

Acceptance Test Cases Cont.

3. Settings Configuration:

- Adjust the number of players using the slider and verify that the displayed value updates accordingly.
- Adjust the starting chips using the slider and verify that the displayed value updates accordingly.
- Adjust the small blind using the slider and verify that the displayed value updates accordingly.
- Click on the "Back" button and verify that the game returns to the main menu with the updated settings.





Acceptance Test Cases Cont.

4. Game Initialization with Settings:

- Start a game from the main menu and verify that it initializes with the correct settings (number of players, starting chips, small blind).

5. In-Game Actions:

- During gameplay, click on the "Check" button and verify that the player's action is processed correctly.
- Click on the "Call" button and verify that the player's action is processed correctly.
- Enter a bet amount in the textbox and click on the "Bet" button, then verify that the player's bet is processed correctly.
- Click on the "Fold" button and verify that the player's action is processed correctly.

6. Game Progression:

- Test the progression of the game through different stages (pre-flop, flop, turn, river) and verify that it advances correctly.
- Verify that the game updates the pot size and player actions accordingly during each stage.


Andrew_AI
Chips: 1000
BET: 0
BET + 1
Leonard_AI
Chips: 999
BET: 1
POT: 3
Hector_AI
Chips: 1000
BET: 0
BET + 2
Thomas_AI
Chips: 998
BET: 2
Pre-Flop
Dealer
Button
PLAYER
Chips: 1000
BET: 0
Back
Call
Bet
Fold


BET + 0
Andrew_AI
Chips: 900
BET: 100
FOLDED
Leonard_AI
Chips: 999
BET: 1
POT: 577
Hector_AI
Chips: 900
BET: 100
BET + 176
Thomas_AI
Chips: 724
BET: 276
Flop
Dealer
Button
BET + 0
PLAYER
Chips: 900
BET: 100
Back
100
Call
Bet
Fold


FOLDED
Andrew_AI
Chips: 900
BET: 100
FOLDED
Leonard_AI
Chips: 999
BET: 1
POT: 753
FOLDED
Hector_AI
Chips: 900
BET: 100
Thomas_AI
Chips: 724
BET: 276
Turn
Dealer
Button
PLAYER
Chips: 724
BET: 276
Back
100
Check
Call
Bet
Fold


PLAYER WINS!
FOLDED
Andrew_AI
Chips: 900
BET: 100
FOLDED
Leonard_AI
Chips: 999
BET: 1
POT: 0
FOLDED
Hector_AI
Chips: 900
BET: 100
WINNER
CALL
Thomas_AI
Chips: 624
BET: 376
Showdown
Reset
Dealer
Button
BET + 100
PLAYER
Chips: 1577
BET: 376
Back
100
Check
Call
Bet
Fold

Acceptance Test Cases Cont.

7. Winning Conditions:

- Test scenarios where a player wins the round (by having the best hand or all other players folding).
- Verify that the game correctly identifies the winner(s) and distributes the pot accordingly.

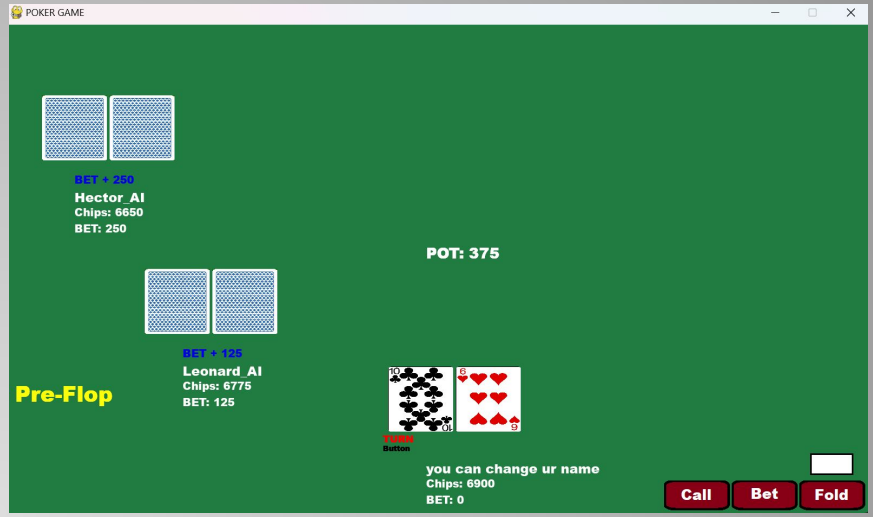
8. Game Termination:

- Test quitting the game from the main menu and in-game.
- Verify that the game terminates without errors and returns to the desktop.

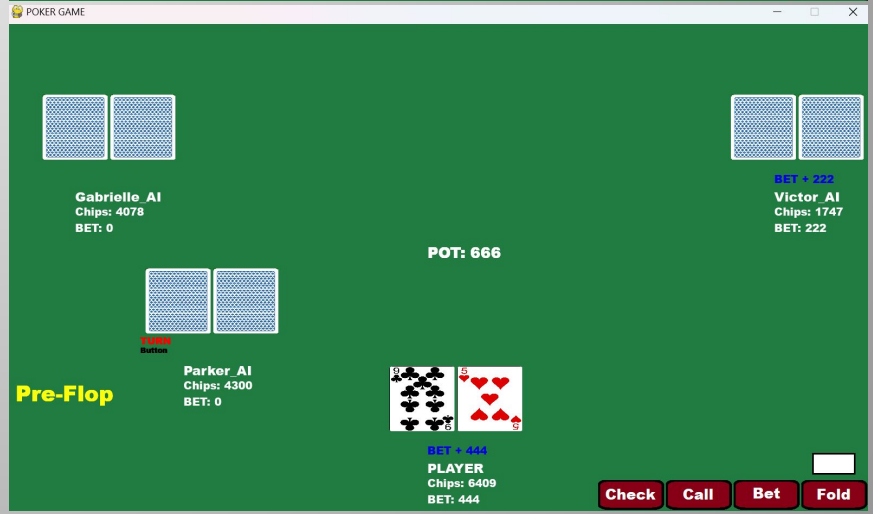
9. Performance:

- Test the game's performance under various conditions (e.g., different screen resolutions, number of players).
- Ensure that the game maintains a stable frame rate and responsiveness.

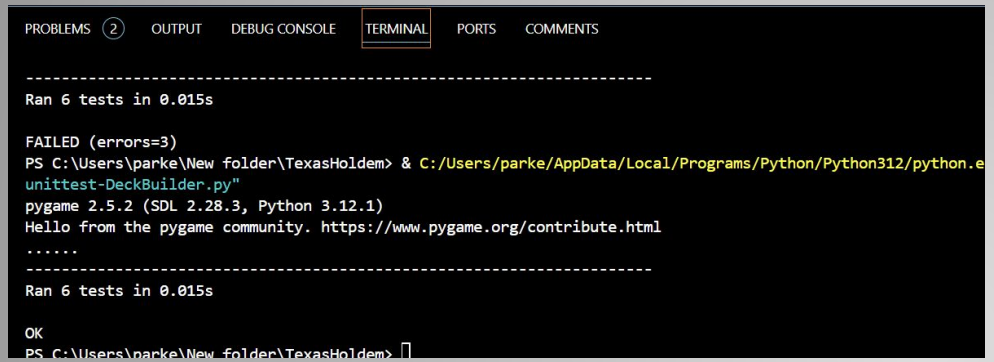

POKER GAME
BET + 2
Oscar_AI
Chips: 1408
BET: 2
BET + 2
Rachel_AI
Chips: 997
BET: 2
POT: 87
Button
BET + 2
Beatrice_AI
Chips: 724
BET: 2
BET + 79
Karen_AI
Chips: 786
BET: 79
Pre-Flop
BET + 2
PLAYER
Chips: 998
BET: 2
Call
Bet
Fold
FAILED (errors=6)
main.py"
Hello from the pygame
Textbox clicked!
Textbox clicked!
Textbox clicked!
Textbox clicked!
Invalid bet amount!
Invalid bet amount!
Invalid bet amount!

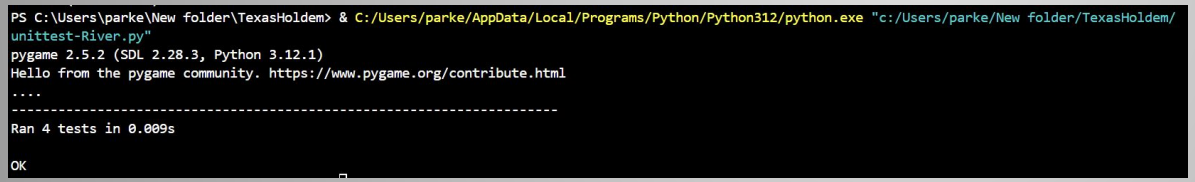

POKER GAME
BET + 250
Hector_AI
Chips: 6650
BET: 250
POT: 375
BET + 125
Leonard_AI
Chips: 6775
BET: 125
Pre-Flop
Button
you can change ur name
Chips: 6900
BET: 0
Call
Bet
Fold

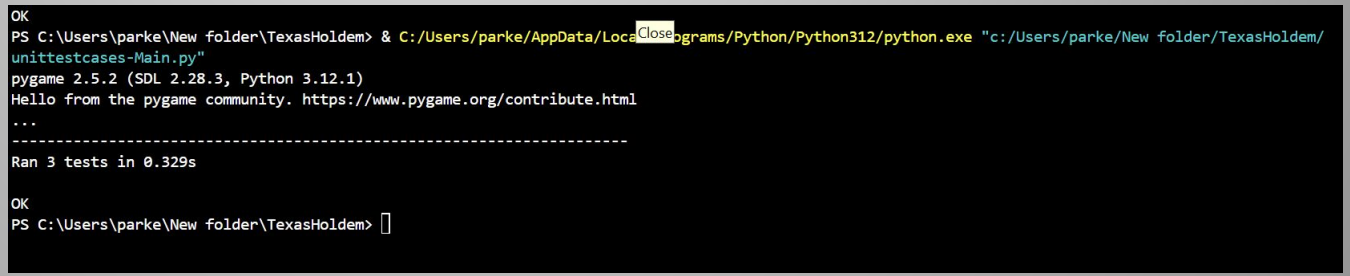

POKER GAME
PLAYER WINS!
FOLDED
Gabrielle_AI
Chips: 4078
BET: 222
CALL
Victor_AI
Chips: 1969
BET: 2331
POT: 0
FOLDED
Parker_AI
Chips: 4300
BET: 0
WINNER
Button
CALL
PLAYER
Chips: 6853
BET: 2331
Showdown
Reset
Check
Call
Bet
Fold


POKER GAME
Gabrielle_AI
Chips: 4078
BET: 0
BET + 222
Victor_AI
Chips: 1747
BET: 222
POT: 666
Button
Parker_AI
Chips: 4300
BET: 0
Pre-Flop
BET + 444
PLAYER
Chips: 6409
BET: 444
Check
Call
Bet
Fold

Unit Testcases(Screenshots)







Git Repo Screenshots

The team used github for Version Control.
Link: https://github.com/aewhitfield90/TexasHoldem
Members used other desktop tools as well such as GitKraken.