**DIAGRAM**



Main Menu: The initial menu when the game first begins. Used to navigate to subcomponents.

Rules: The printouts of the rules of this game.

Settings: The generic settings for the game to play.

Game: The main engine for the Blackjack game. This includes the UI.

Player: The base player class. Includes member variables for things such as player name, chip count, hand value etc.

CPU: Similar to player, however instead instead of being given the option to make a move, obtains the next move from predetermined lookup tables.

Lookup Tables: the files that hold the 2D matrices CPU’s use to determine their next move

Card: This class to stores information and methods relating to the cards in the blackjack game.

We chose to perform incremental testing with the top-down approach. We chose to do this because our game follows a very sequential path, so in order to reach the bottom modules such as card and lookup tables, you first have to go through other modules. This mirrors normal gameplay.

|  |  |
| --- | --- |
| **Module** | **Developer** |
| Main Menu | Trevor, Sean |
| Rules | Trevor |
| Settings | Trevor |
| Game | Sean, Dan, Scott, Trevor |
| CPU | Dan, Sean |
| Player | Dan, Scott |
| Lookup Tables | Dan |
| Card | Scott |

|  |  |
| --- | --- |
| **Module** | Component [1] - Main Menu |
| **Input** | Integer between 1-4 inclusive |
| **Output** | Rules, Menu for Settings, Menu for Game, Exit message, Invalid Input Warning |
| **Dependencies** | Rules, Game, Settings |

|  |
| --- |
| **Incremental Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | Input: ‘3’ -> ‘return’  Attempt to go to settings then immediately return to main menu, infinite loop into stack overflow error occurs | 3 | Attempted fix at previous settings page bug. The settings would previously take in additional input, which could result in undesirable behavior. Fixed by reverting change. (Removed creation of new scanner for settings method) |

|  |
| --- |
| **Regression Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No further bugs found | n/a | n/a |

|  |  |
| --- | --- |
| **Module** | Component [2] - Rules |
| **Input** | n/a |
| **Output** | Text containing the rules of the game then the main menu |
| **Dependencies** | none |

|  |
| --- |
| **Incremental Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No further bugs found | n/a | n/a |

|  |
| --- |
| **Regression Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No further bugs found | n/a | n/a |

|  |  |
| --- | --- |
| **Module** | Component [3] - Engine |
| **Input** | Integer 1-4 inclusive to select action, Integer 1-[MAX\_CHIPS] inclusive to bet chips at beginning, 4 being the “split” option |
| **Output** | Hands of players in ASCII, Game menu, |
| **Dependencies** | CPU, Player, Card |

|  |
| --- |
| **Incremental Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | Upon end game condition where either all players have busted or are staying, final hands of players are not printed | 2 | Call printBoard() before checking end game conditions |
| 2 | AI is currently under development. Game breaking errors still occur during play | 3 | Finish AI and bug test |
| 3 | All players still able to select move option on their turn even if they have previously busted | 2 | Check for bust before player gets to decide their move |
| 4 | After player has busted or won the round, selecting Continue or Quit Game options have no effect | 1 | Ensure correct function is called with correct parameters set |
| 5 | On win condition, specific players not notified of their winnings, no clear winner | 2 | Print final hand, bets, and winnings of all players in a well-organized manner |
| 6 | On end game condition where human busts, all CPU choose to stay, game stuck in infinite loop of not selecting winner and CPU selecting stay | 1 | Method to determine is the round is over was changed to properly determine the winner of the CPUs who haven’t busted |
| 7 | On end game condition where all players choose to stay, no winner declared | 1 | Method to determine who wins the hand changed if condition to properly determine the winners if nobody busts |
| 8 | On end game condition where all players choose to stay, if player selects ‘Hit’ following turn, null pointer exception at insertCard() | 1 | Player can no longer perform a hit or stay action if they have already hit or stayed, as their hand is done for the round. All other players proceed with their hand until the round is over. |
| 9 | If human player busts and CPU’s choose to stay, once game ends, human player has no option to restart a new game, only Continue or Quit | 2 | Change menu options based off current game conditions |

|  |
| --- |
| **Regression Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | When dealing with aces hand value was off by one due to a misformed if/else if/else | 2 | changed the second if statement to an else if statement |
| 2 | Upon fixing [4] above, game not reset on an end game condition, automatic win for all players of previous game | 1 | Reset all game variables to defaults once game ends and clear winners list |

|  |  |
| --- | --- |
| **Module** | Component [4] - Settings |
| **Input** | the strings “chips”, “cpus”, “return”, “difficulty” followed by an integer on a new line. (Used strings instead of of number input to eliminate extra dialogue and confusion) |
| **Output** | Confirmation of the changing of the setting variable |
| **Dependencies** | Main Menu |

|  |
| --- |
| **Incremental Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | Settings does not print a string to confirm changes | 3 | Add print statement |
| 2 | The input of a keyword without an integer following causes out of bounds exception and crash | 3 | Add check statement |

|  |
| --- |
| **Regression Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | Settings page prints changes despite it failing to have made any changes as a result from fixing incremental test #1 | 3 | Move print statement into else statement so that it only prints when successful |
| 2 | Settings page will print two error strings when inputting a keyword without a trailing integer | 3 | Move try catch statement into if statement so it only executes upon proper input, not just when keyword is found |

|  |  |
| --- | --- |
| **Module** | Component [5] - Player |
| **Input** | n/a |
| **Output** | Actions such as hit, split, stay, hand value |
| **Dependencies** | Card, Engine |

|  |
| --- |
| **Incremental Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | hand value is not calculated properly. Aces are always viewed as 1 point instead of possibly 11 and face cards are viewed as 11- 13 points respectively | 1 | Add extra check statement for aces and face cards |

|  |
| --- |
| **Regression Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No further errors found | 0 | n/a |

|  |  |
| --- | --- |
| **Module** | Component [6] - CPU |
| **Input** | Internal system call to play computer turns |
| **Output** | Confirmation that CPU made a move |
| **Dependencies** | Lookup Tables, Card, Engine |

|  |
| --- |
| **Incremental Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | Out of bounds error when the CPU had an ace as one of their cards | 1 | Changed the access index of the softTables to match that of the respective card |

|  |
| --- |
| **Regression Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | Would continually stay instead of sometimes hitting | 1 | Not a problem with lookup table, error in revised switch statement. Boolean logic was incorrect, changed to correct statement. Adequately calls hit method when supposed to now. |

|  |  |
| --- | --- |
| **Module** | Component [7] - Card |
| **Input** | Random Seed |
| **Output** | Suit, Value |
| **Dependencies** | None |

|  |
| --- |
| **Incremental Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | Ace value should either be 1 or 11 but it is only 11 | 2 | Add a check to see if the card is an ace, give it a special flag if it is |

|  |
| --- |
| **Regression Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No further errors detected | 0 | n/a |

|  |  |
| --- | --- |
| **Module** | Component [8] - Lookup Tables |
| **Input** | None |
| **Output** | Decisions for the AI |
| **Dependencies** | None |

|  |
| --- |
| **Incremental Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No further errors detected | 0 | n/a |

|  |
| --- |
| **Regression Testing** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No further errors detected | 0 | n/a |