**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 | No Bugs Found | N/A | N/A |

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No 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 Bugs Found | N/A | N/A |

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No 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 successful end-game condition, board is printed twice (CPU hands twice, player hand once) | 3 | Removed double print statement in playGame method |
| 2 | Upon successful end-game condition, in the case that player has 0 chips left and chooses to start new game, infinite loop where player is asked to bet chips occurs | 1 | Added check for 0 chips when new round starts, will reset chip amount to default and print message |
| 3 | When ‘3’ followed by pressing the enter button is repeated fast enough, messages are “queued up” and consequent actions cause them to be printed after every other action until the queue has emptied | 3 | No way to solve, problem lies directly with Java print statement handling. It is caused by the while loop checking for input completing faster than the print statements |
| 4 | Upon player busting, games does not end. Player must ‘Stay’ until computers finish their moves | 2 | Create method that will execute CPU turns until each CPU chooses to stay or busts, then determine winners |
| 5 | Inputting an integer out of range does not prompt an error message | 3 | Added print statement to warn players of proper input |
| 6 | Player can still hit after staying once | 1 | The round should end upon the player staying. Fix by adding a call to the method that plays all CPUs to end created for defect #4 |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | After fixing Defect #2 in Incremental, noticed that the player does not see chip count after it is reset to default | 3 | Added redundant print statement in case of chip reset to display new chip count |
| 2 |  |  |  |

|  |  |
| --- | --- |
| **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 | No Bugs Found | N/A | N/A |

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No Bugs Found | N/A | N/A |

|  |  |
| --- | --- |
| **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 | When player has busted or stayed, they have to continually select “stay”, even after they are out of moves | 3 | Automatically end the player’s turn if they cannot perform any actions to avoid unnecessary dialogue |

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | When automatically ending the human player’s turn after they stay,, player remains in “stay” state for the next round | 1 | After a round is over, reset player’s “stay” state to false |
| 2 | When automatically ending the human player’s turn after they bust, player remains in “bust” state for the next round | 1 | After a round is over, reset player’s “bust” state to false |

|  |  |
| --- | --- |
| **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 | No bugs found | N/A | N/A |

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No bugs found | N/A | N/A |

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

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | CPUs’ first cards were remaining hidden, even after the round was over. This did not allow the human player to see the winning hand. | 3 | Add a condition for the print board method to print all of the cards, including the first CPU cards, given that the round is over. |

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No Bugs Found | N/A | 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 bugs found | N/A | N/A |

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect #** | **Description** | **Severity** | **How To Correct** |
| 1 | No bugs found | N/A | N/A |