**Group Project**

**COSC 1174**

**Due 11/21/20**

**Jokers Wild!**

**Grading Breakdown**

Video Presentation 30%

Self/Peer Assessment 30%

Completeness of Program 30%

UML/Flow Diagram/README 10%

**Group Members**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_Dillon Olbrich\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Nathan Ha\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Rubric**

Video Presentation

One or more members of the group need to record themselves explaining the program in detail, while going over the code. Then, they need to run the program, and play the game enough to show all the functions behave the way they intended. If anything doesn’t work, they need to explain what the group couldn’t figure out, and how they worked around it. The video does not need to include all group members, and only screen capture with voiceover is necessary.

Self/Peer Assessment

At the end of this document is the Self/Peer evaluation. Please provide estimations of each member’s contribution to each category, as well as a brief description of your personal contribution. Also give every member (including yourself) a grade, based on how you think the project turned out, and how well that person contributed to making that grade.

Completeness of Program

The following page describes all the features required for the game. The more things you have, the better the grade, but this will also scale with how many members of the group there are. If smaller groups are able to implement all of the features, I will also consider scores above the 30% listed, as an extra credit bonus.

UML/Flow Diagram/README

The UML should include all classes outside other than the controller (main method) class. The flow diagram should describe how the controller class functions, and how the main method passes control and makes calls to other classes and methods. The README file should explain how you run the program, including if it must be imported to a specific dev software, and what packages or add-ins are needed.

**Joker’s Wild!**

This project will expand on the Poker assignments to a fully playable game. In Poker #3, we added betting to the game and checked for a ‘winning’ hand. Now, we want to include the Jokers, which can count as any card in the deck. We also need to add tiered payouts, so better hands pay more, i.e. Full House wins more money than a pair. With the introduction of wilds, there is a different table of odds you’ll want to look up, for instance a single pair is no longer a winning hand. With that said, here is a list of features to consider. A group of 6 should look to implement all of them, while a group of 2 need only pick 2 for full credit.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Group Size | 2 | 3 | 4 | 5 | 6 |
| Minimum Features Needed | 2 | 4 | 6 | 8 | 10 |

* Animation for ‘big’ wins
* A radio button option that changes the number of decks being shuffled
* An internal trigger that reshuffles the decks when less than 25% of the cards are left
* A shuffling animation that uses the back of the cards
* Upgrade the betting options to a textField that lets you bet up to all of your money
* Bet 1 and Bet Max buttons that automatically fill in the bet textField when clicked
* Remove the hold buttons, and move the function to clicking on the cards themselves
* Change the option from ‘hold’ to ‘discard’
* When a card is selected for discard, it is flipped face down
* Options for at least 6 different avatars in the top left corner that scroll through the options when clicked

**Self/Peer Assessment**

Fill the table with the member names and approximate percentage of their contribution in each category. Below the table, specify what your contribution was.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Member Name | Video | Program | Documentation | Total |
| Dillon Olbrich | 50 | 50 | 50 | 150/150 |
| Nathan Ha | 50 | 50 | 50 | 150/150 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Nathan and I worked on the program and got ideas from each other and pieced it together. I recorded the video and Nathan wrote a script for me to go over. As for the documents The documentation we both did with passing it back and forth and editing it. Nathan is a great partner to work with for both semesters. Equal effort both ways.