Final Project Milestone 2

Project Implementation

Course: DGL-204

Student: Reeve Jarvis

Instructor: Brad Best

Submit Date: 12/06/2021

As detailed in my milestone 1 report, my final project was intended to be a “gameroom” application that held the backend logic required to play three games. This idea proved to be too substantial for me to implement as I had intended. I have therefore restructured and refocussed towards a much more simplified version of this concept, with room to build on it in the future. I set some very high goals for this assignment, as I usually tend to do, but I suppose that is all part of the process of idea generation and implementation.

My overall goal was to provide a functional and usable backend that would allow for a guided player experience. The user could select from single or two player modes, pick a game, and then continue to play that game until the decision to quit or a winner was found. Additionally, I had intended to include logic for checkers and uno. The final submitted version of this project is lacking many of these features, but I have still placed some significant effort into producing something that can be built upon and improved in the future.

My intention for this project was to focus it on a topic that is very close to my heart, gaming. I wanted to explore the various moving pieces and logical components required to create a gameplay experience. I feel like I was still able to accomplish this goal, even if my project is incomplete. I certainly gained some insight into how large a task it is to make a game work. There are so many moving parts to keep track of, even in a game as simple as Yahtzee. I may have bitten off more than I can chew with this project idea.

In its current form, my project holds most of the required back-end logic needed to play a game of Yahtzee, with some notable room for improvement. I have built the various game objects required using classes, structures, and enumerations. Additionally, I have created many methods/functions to handle the processing of information and the navigation through the gameplay loop. These game objects have various stored and computed properties in order to hold data values relevant to the game. I used tuples in cases where I needed to track multiple values (Example: I used tuples in my scoreboard structure to track the fields names, their value, and whether they had been tracked yet). I also used enumerations in order to limit the number of valid choices for die selections and scorecard fields. This assisted me in creating the logic for player decisions. Additionally, it served as a form of input validation, though I was unable to fully implement error handling as I would have liked. I tried my best to use iteration and other concepts to reduce repetitiveness, but I am aware that there is much more room for improvement in this sense.

At the bottom of my project, I included a small “testing” or more like “code-exercising” section where I have added some documentation regarding the expected usage of my code. This section is meant to serve as an example of how a gameplay loop may work in a game of Yahtzee, and to show the results of my work. I wanted this to be a more interactive experience in leu of any actual interaction. I have left some code statements to fill in, with comments indicating instructions and the choices available. I did this to replicate the decisions that may be included in the “user-interface” that may one day go along with this back-end logic. Results are shown in the console using print statements.

In conclusion, I did my best to produce something to submit that leverages many of the concepts we have covered in this course. I know that this is certainly not the best code I have written and have marked myself accordingly on my rubric. I just hope that my efforts are evident in my work.