SYSC 3110

Software Development Project

Project Report

UNO FLIP

Submitted by

TEAM 28

Nikita Vijay – 101195009
Samson Ha- 101231965
Mofipefoluwa Olatunji – 101237522
Keya Patel - 101191689

October 22, 2023

Carleton University,
Faculty of Engineering and Design

1 Contents

2 The Problem Statement:

The official game of UNO has been a game that's being played for a while. With its various versions and iterations UNO Flip was introduced with a double-sided deck card that consisted of a dark and light side. The key was to use the flip card to switch the side of the deck being played by the users.

UNO requires more than 2 players to be played, but what happens when you do not have a player to play with? UNO came up with the idea of making the game digital and accessible to everyone.

UNO can now also be played on various online gaming websites with other players all across the world.

3 The Project Goal

The goal of this project is to create a MVC pattern-based UNO flip gaming program. The program will be developed to play on a desktop using Java as the programming language. The game will start as a text-based game and build up to the final milestone which involves using various java GUI classes, along with implementing the MVC pattern for the game interface.

The goal of the project is to learn about the various design patterns such as decoupling, code refactor, inheritance and various other Java based implementations.

4 The Project Design:

The project consists of four milestones, by adding functionalities on each milestone, making the game more advance on each milestone.

During the first iteration, a text-based playable version of the game where players can play the game via the console. The game takes 2-4 players and assigns them a hand. Each player must follow the UNO flip rule by placing and picking up cards as per the official UNO flip game notation.

The system design consists of the main java class called Game.java. The main() method prompts the player to enter the number of players playing and prompts the user for the names of each player.

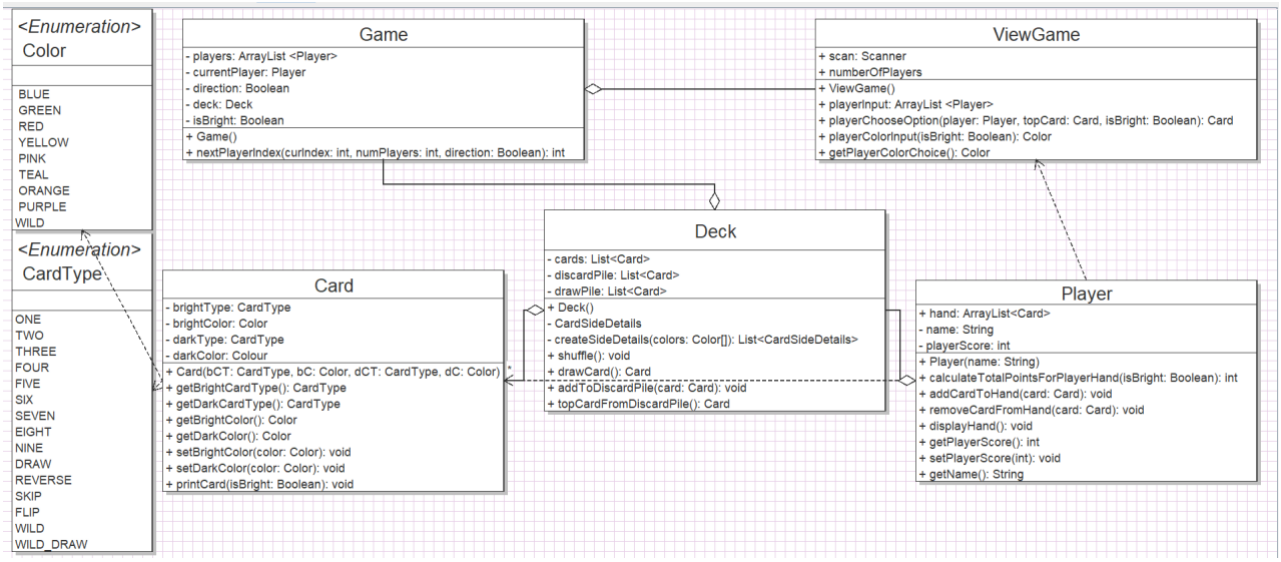
Milestone 2 requires the programmer to implement the text base game and convert it to GUI using the MVC pattern implementation and have a user interact with the GUI based version.

Milestone 3 requires the programmer to implement the GUI using AI players.

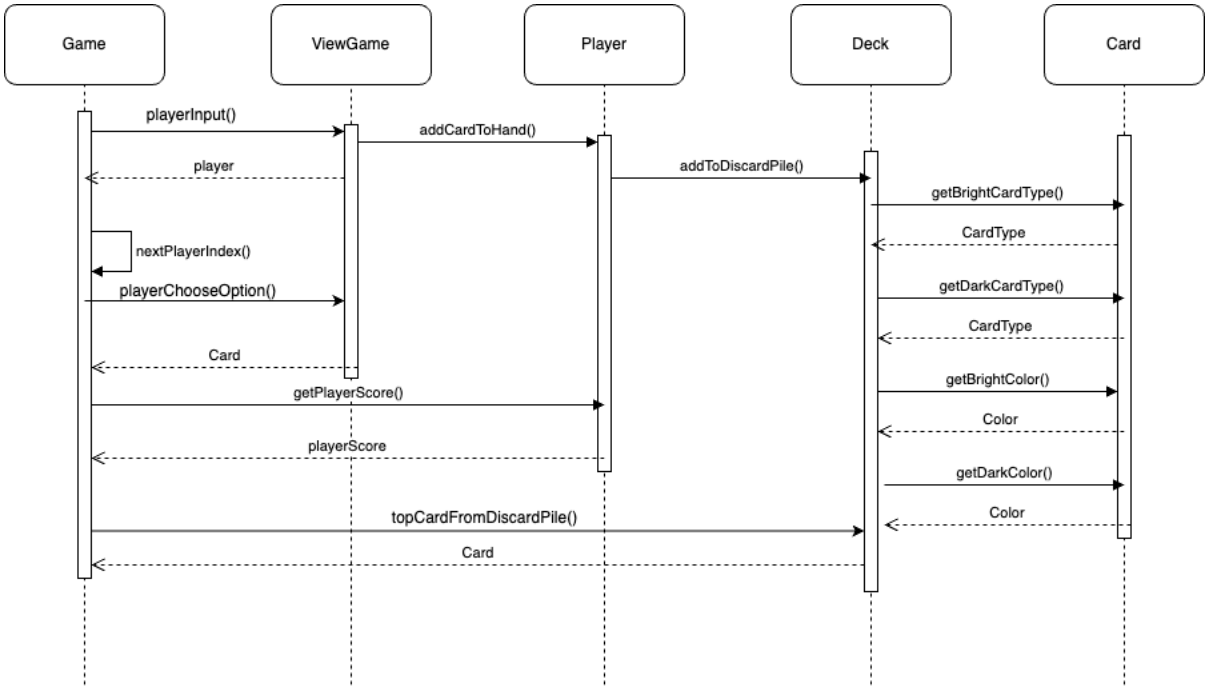
Milestone 4 requires the programmer to add functionalities such as back, save, undo and etc to the game.

5 UML DESIGN

CLASS DIAGRAM



SEQUENCE DIAGRAM



Data Structure Explanation:

Game.java uses an arrayList players of type Player. We have used arrayList as the data structure as it becomes easier to add and removes players. Moreover, the task of assigning a hand to each player also makes it easier by calling on the arrayList.

Deck.java uses two arraylists called cards and discardPile of type Card. The cards arrayList acts as a deck of cards in UNO. Implementing functions such as cards.add(new card(Light, color, Dark, color)) becomes eaiser to use. Furthermore, removing a card from the deck is easy by using the function cards.remove(0), which is the card that the user picks up from the top of the deck. Similar to the cards arraylist, the discardPile is used to discard a card from the users hand, by using discardPile.add(card), which adds the card that is removed from the hand to the discard pile. Furthermore, it also serves the purpose of checking the last card entered and validating the card discardPile.get(discardPile.size()-1).

Player.java uses an arraylist called hand which is of type Card. This is the hand of each player and the cards they have in hand while playing the game. ArrayList makes it easier to add a card to the hand by using hand.add(card) or removing a card from the players hand by calling hand.remove(card).

6 User Manual

Milestone 1 Steps to play the game

Step 1: Run the main() method in class Game.java. The console will prompt the player to enter the number of players

Step 2: Once the number of players are entered, the user will be prompted to

add their names

Step 3: Select a number based on the card that needs to be played from the deck into the discard pile.

7 REFERENCES