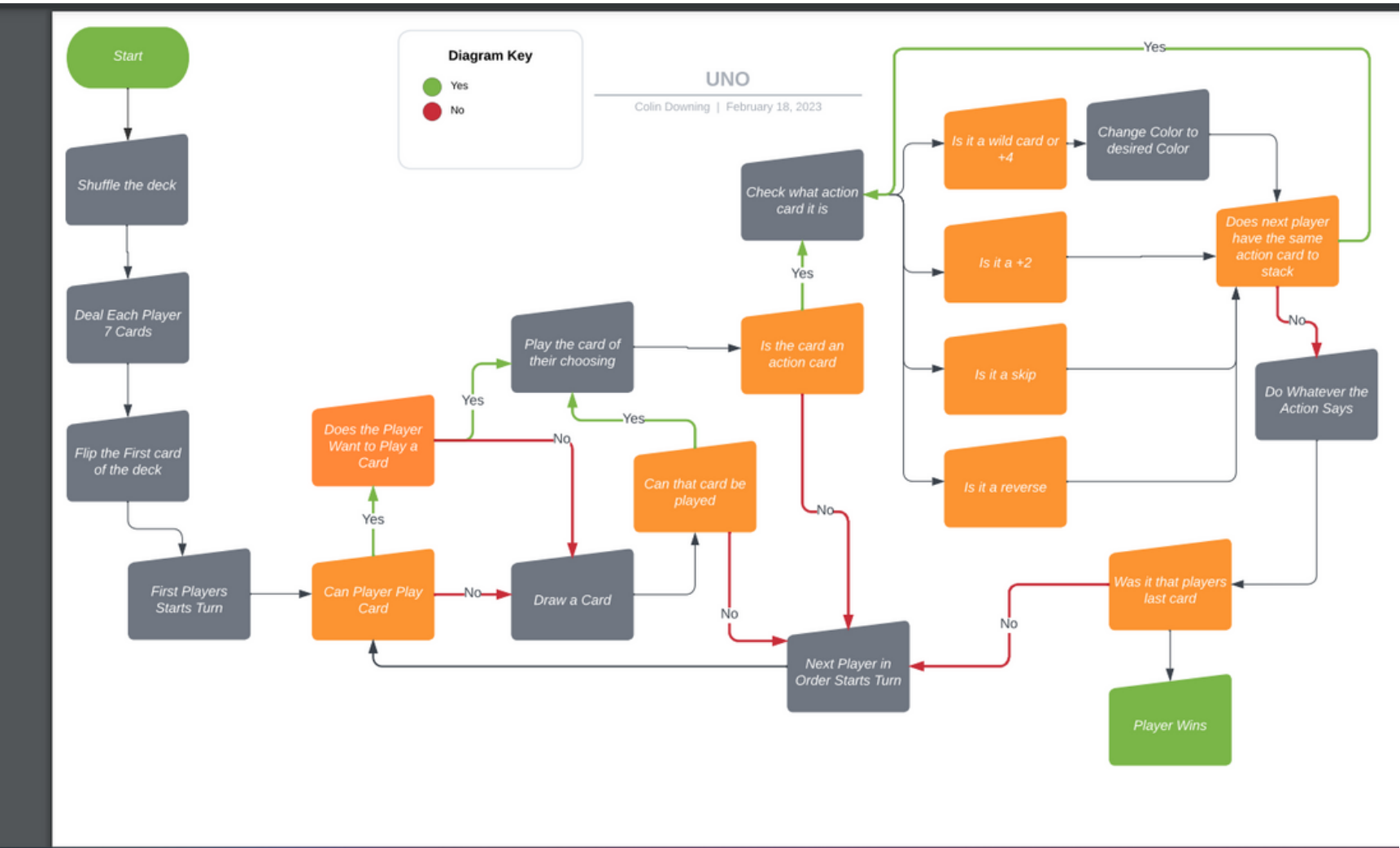


# Milestone 1

In Milestone 1 I designed my Storyboard and my flowchart to help give me a basic baseline to build my code off of for the future steps.



# Milestone 2

In milestone 2 I began to develop my classes and variables to use for the code later in the project

```
public class Deck {
    protected Card topCard = new Card();
    protected int cardsInDeck;

    public Deck(Card topCard, int cardsInDeck) {
        this.topCard = topCard;
        this.cardsInDeck = cardsInDeck;
    }

    public Deck() {
    }

    public Card getTopCard() {
        return topCard;
    }

    public int getCardsInDeck() {
        return cardsInDeck;
    }

    public void setCardsInDeck(int cardsInDeck) {
        this.cardsInDeck = cardsInDeck;
    }
}
```

```
public class Player {
    protected int cardsInHand;

    public Player(int cardsInHand) {
        this.cardsInHand = cardsInHand;
    }

    public Player() {
    }

    public int getCardsInHand() {
        return cardsInHand;
    }

    public void setCardsInHand(int cardsInHand) {
        this.cardsInHand = cardsInHand;
    }

    public void drawCard(Deck d) {
        this.setCardsInHand(this.getCardsInHand()+1);
        d.setCardsInDeck(d.getCardsInDeck()-1);
    }

    public void playCard(Stack s, Card c) {
        this.setCardsInHand(this.getCardsInHand()-1);
        s.setTopCard(c);
    }
}
```

```
public Card(int value, String color) {
    this.value = value;
    this.color = color;
}

public Card() {
}

public int getValue() {
    return value;
}

public void setValue(int value) {
    this.value = value;
}

public String getColor() {
    return color;
}

public void setColor(String color) {
    this.color = color;
}

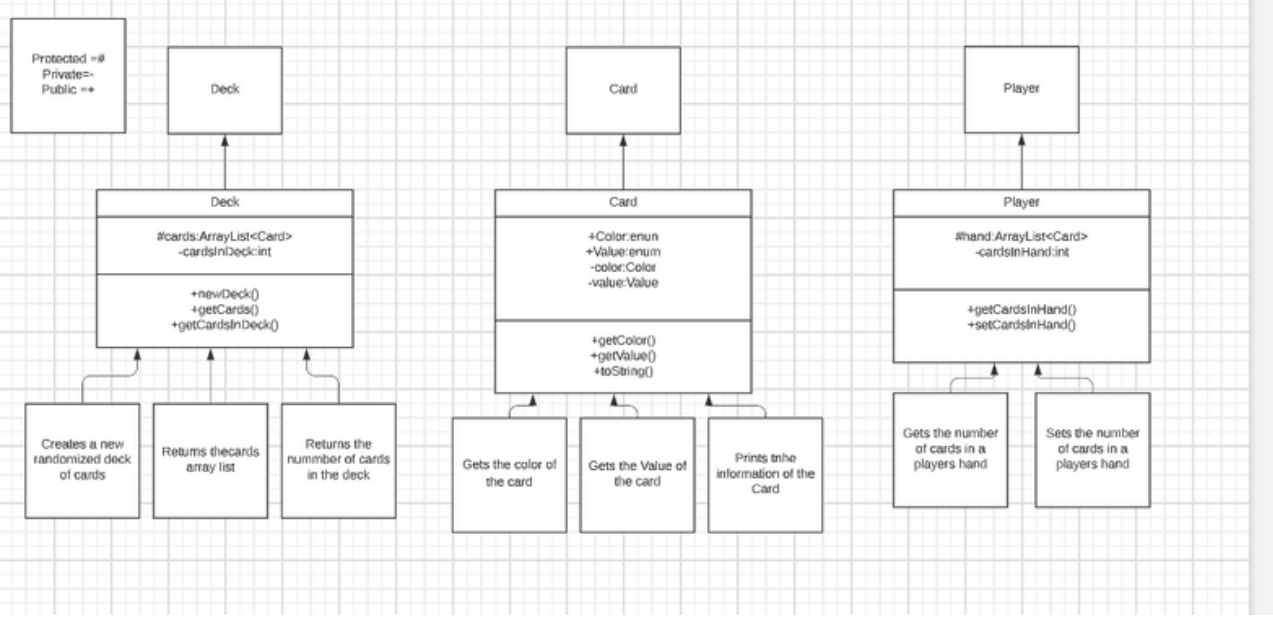
public String getAction() {
    return action;
}

public void setAction(String action) {
    this.action = action;
}

public boolean isMatch(Card c1, Card c2) {
    if(c1.getValue()==c2.getValue() || c1.getColor()==c2.getColor())
        return true;
    else
        return false;
}
```

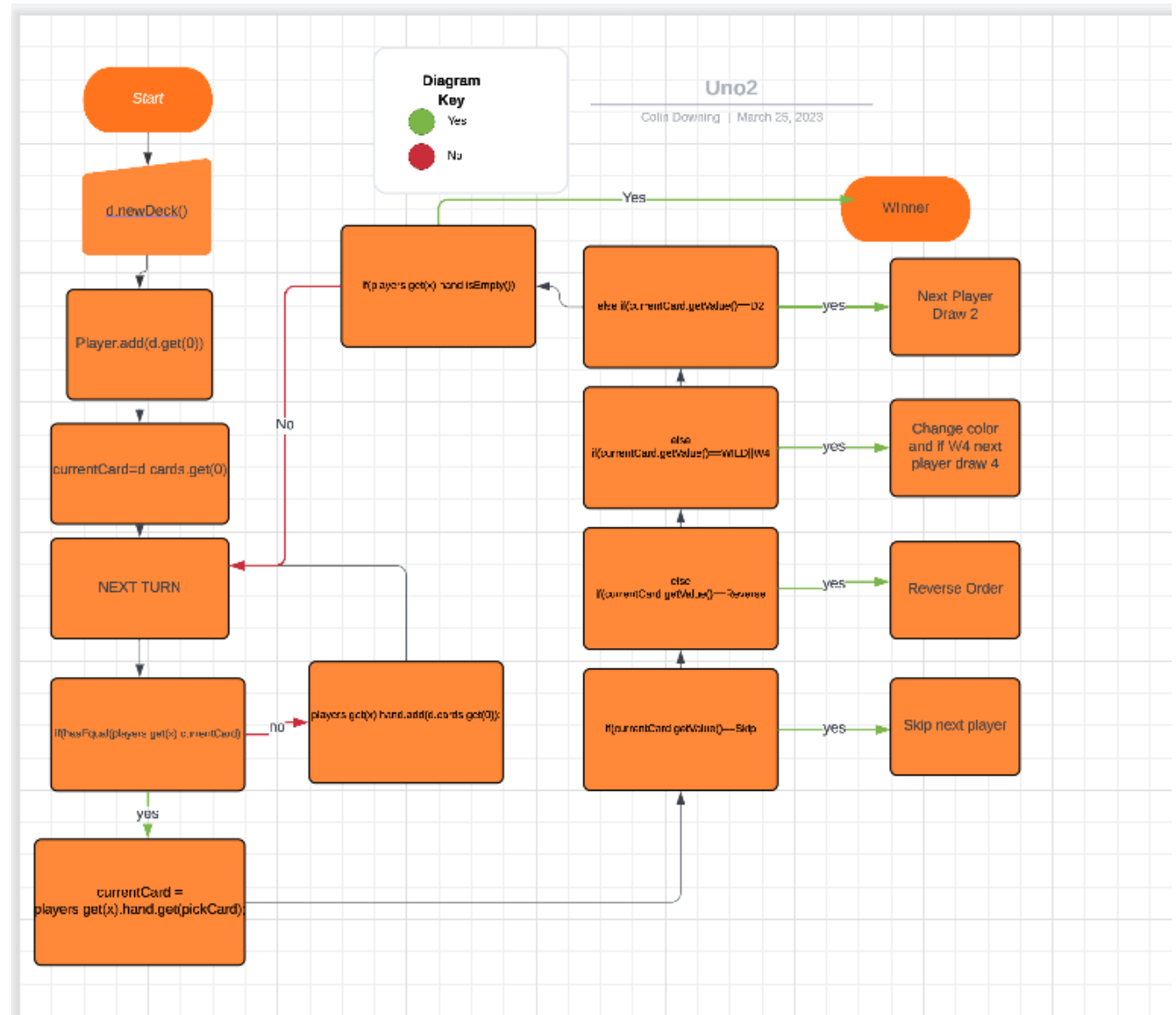
# Milestone 3

In milestone three I made my UML Diagram and finished designing my classes and added more methods



# Milestone 4

In Milestone 4 I finished writing my code to work in the terminal and reorganized my flowchart to make it use the methods that I wrote



```
"C:\Program Files\Java\jdk-17.0.4\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2022.3.2\lib\idea_rt.jar=56036:C:\Program Files\Java\jdk-17.0.4\bin" -Dfile.encoding=UTF-8
[Card{color=Blue, value=Reverse}, Card{color=Red, value=Two}, Card{color=Red, value=Reverse}, Card{color=Blue, value=Six}, Card{color=Red, value=Zero}, Card{color=Blue, value=Eight}, Card{color=Yellow, value=Eight}]
Pick a card to play or draw a card >>> 1.Play Card 2.Draw Card

You don't have a card to play
Drawing Card
[Card{color=Yellow, value=Nine}, Card{color=Red, value=Seven}, Card{color=Green, value=Skip}, Card{color=Red, value=D2}, Card{color=Blue, value=Five}, Card{color=Yellow, value=Eight}]
Pick a card to play or draw a card >>> 1.Play Card 2.Draw Card

Pick the card to play- enter the number of the card in your hand
[Card{color=Green, value=Eight}, Card{color=Green, value=Nine}, Card{color=Yellow, value=Five}, Card{color=Green, value=Seven}, Card{color=Green, value=One}, Card{color=Yellow, value=Nine}]
Pick a card to play or draw a card >>> 1.Play Card 2.Draw Card
```

# Milestone 5

In Milestone 5 I had many problems trying to get the code I had previously written for the terminal to work in the GUI. I used the Brightspace videos to help me get the GUI t to almost work.

