**Deliverable 3**

Muharrem Kaya

Derya Kaya

Mustafa Soroush

Nil Patel

Prof: Amandeep Sidhu

April 17, 2020

Contents

[1. UML Class Diagram 3](#_Toc38042299)

[2. Referance Git Repository Address 3](#_Toc38042300)

[3. Referance Git Repository Location JUnit Test 3](#_Toc38042301)

[4. Test Instructions 3](#_Toc38042302)

[a. testCheckPlayerCountGood(): 3](#_Toc38042303)

[b. testCheckPlayerCountBad() 3](#_Toc38042304)

[c. testCheckPlayerCountBoundry() 3](#_Toc38042305)

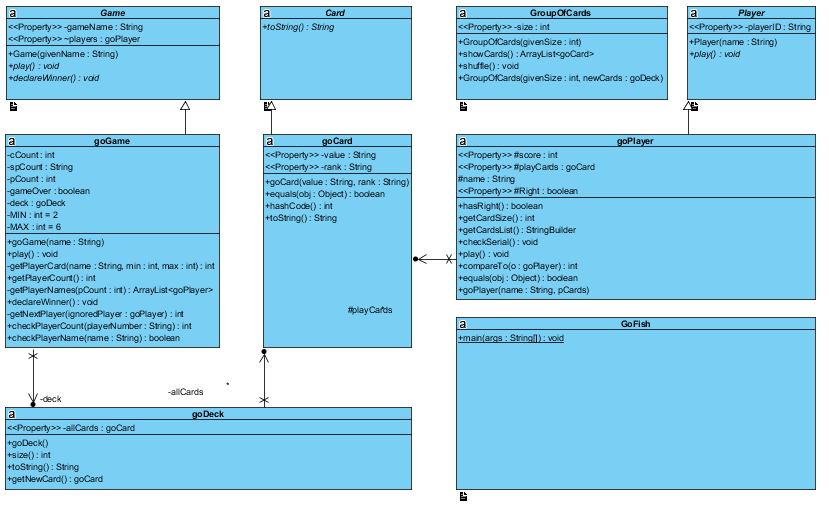
[d. testCheckPlayerNameGood() 4](#_Toc38042306)

[e. testCheckPlayerNameBad() 4](#_Toc38042307)

[f. testCheckPlayerNameBoundry() 4](#_Toc38042308)

[5. Test Results 4](#_Toc38042309)

# UML Class Diagram



# Referance Git Repository Address

https://github.com/mucteb/Deliverable1

# Referance Git Repository Location JUnit Test

https://github.com/mucteb/Deliverable1/tree/master/test/ca/sheridancollege/project

# Test Instructions

The application has 6 different test methods

## testCheckPlayerCountGood()

This method takes playerNumber as a parameter. The parameter must be between 2 and 6 expected result true. We used “5” for players

## testCheckPlayerCountBad()

This method takes playerNumber as a parameter. The parameter must be between 2 and 6 expected result false. We used “1” for players

## testCheckPlayerCountBoundry()

This method takes playerNumber as a parameter. The parameter must be between 2 and 6 expected result true. We used “6” player.

## testCheckPlayerNameGood()

Players name must become at least 5 characters. This method check players names according requirements, and it takes “name” as a parameter. Expected result is true. We used “Muharrem” as a parameter.

## testCheckPlayerNameBad()

Players name must become at least 5 characters. This method check players names according requirements, and it takes “name” as a parameter. Expected result is false. We used “Ali” as a parameter.

## testCheckPlayerNameBoundry()

Players name must become at least 5 characters. This method check players names according requirements, and it takes “name” as a parameter. Expected result is true. We used “Caner” as a parameter.

# Test Results

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement** | **Use Case** | **Test Method (ClassName.methodName)** | **Status (Date)** |
| Test the eligible number of players. | “check a good number of players” | testCheckPlayerCountGood() | Pass(April 16, 2020) |
| Test the not eligible number of players. | “check a bad number of players” | testCheckPlayerCountBad() | Pass(April 16, 2020) |
| Test the boundary number of players | “check a boundary number of players” | testCheckPlayerCountBoundry() | Pass(April 16, 2020) |
| test an eligible player name. | “check a good player name” | testCheckPlayerNameGood() | Pass(April 16, 2020) |
| test a not eligible player name. | “check a bad player name” | testCheckPlayerNameBad() | Pass(April 16, 2020) |
| test a boundary player name. | “check a boundary player name” | testCheckPlayerNameBoundry() | Pass(April 16, 2020) |

