Group3 Deliverable 1

Group members

Elisha Nesci
Furkan Gonul
Omama Eiz Eddin
Wei-Hsiang Chang

SYST17796 Fundamentals of Software Design and Development

Professor Paul Bonenfant

Feb.2021

Table of Contents

Team Contract

Team Name: Group 3

Please negotiate, sign, scan and include as the first section in your Deliverable 1.

By signing this contract, we acknowledge having read the Sheridan Academic Integrity Policy as

Team Member Names (Please Print)	Signatures	Student ID
Project Leader: Elisha Nesci	EN	991629115
Omama Eiz Eddin	OEE	991627127
Wei-Hsiang Chang	WHC	991623950
Furkan Gonul	FG	991623631

per the link below.
https://policy.sheridanc.on.ca/dotNet/documents/?docid=917&mode=view

Responsibilities of the Project Leader include:

- Assigning tasks to other team members, including self, in a fair and equitable manner.
- Ensuring work is completed with accuracy, completeness and timeliness.
- Planning for task completion to ensure timelines are met
- Any other duties as deemed necessary for project completion

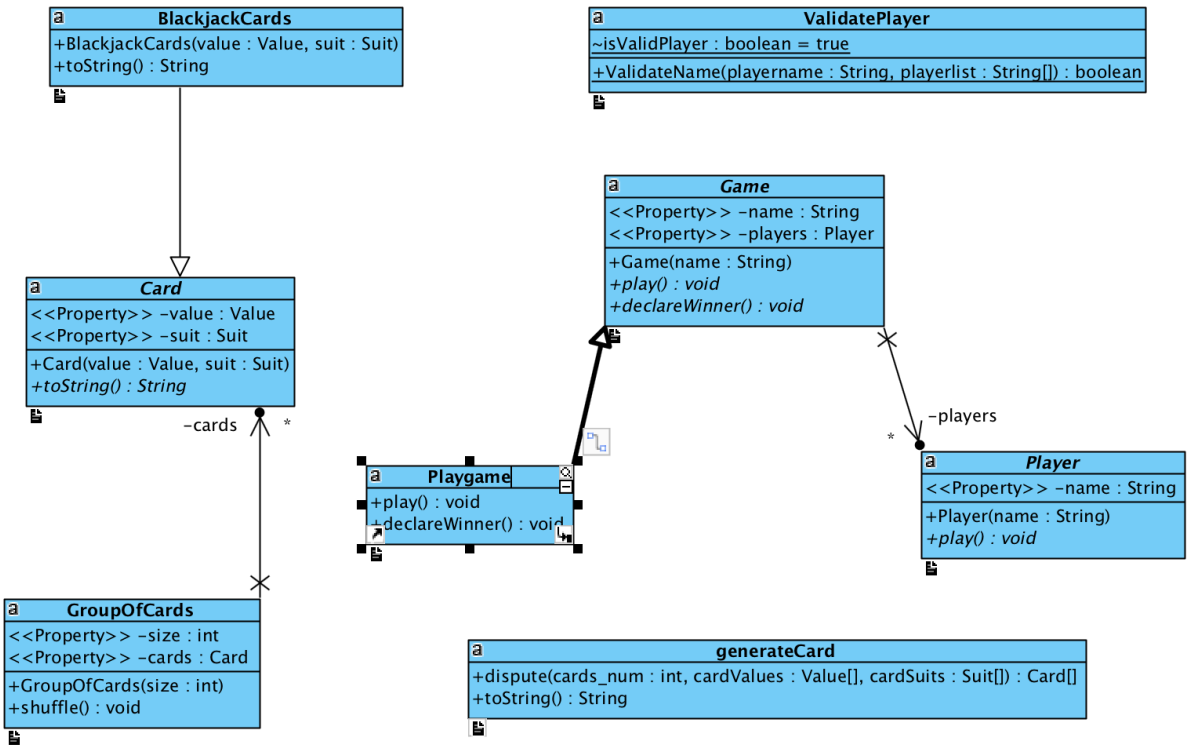
What we will do if . . .

Scenario	Accepted initials	We agree to do the following
Team member does not deliver component on time due to severe illness or extreme personal problem	EN OEE WHC FG	a) Team absorbs workload temporarily X b) Team seeks advice from professor __ c) Team shifts target date if possible __ d) Other:
Team member cannot deliver component on time due to lack of ability	EN OEE WHC FG	a) Team reassigns component __ b) Team helps member X c) Team member must ask professor for reference material __ d) Other:
Team member does not deliver component on time due to lack of effort	EN OEE WHC FG	a) Team absorbs workload __ b) Team "fires" team member by not permitting his/her name on submission __ c) Other: Team will remind her/him of responsibility to complete task X
Team member does not attend team meeting	EN OEE WHC FG	a) Team proceeds without him/her and will assign work to the absent member X b) Team doesn't proceed and records team member's absence __ c) Team proceeds for that meeting but "fires" member after __ occurrences __
An unforeseen constraint occurs after the deliverable has been allocated and	EN OEE	a) Team meets and reschedules deliverable X

Scenario	Accepted initials	We agree to do the following
scheduled (a surprise test or assignment)	WHC FG	b) Team will cope with constraint __ c) Other:
Team cannot achieve consensus leaving one member feeling "railroaded", "ignored", or "frustrated" with a decision which affects all parties	EN OEE WHC FG	a) Team agrees to abide by majority vote X **First (if tie)** b) Team flips coin Second X c) Other:
Team members do not share expectations for grade desired	EN OEE WHC FG	a) Team will elect one person as "standards-bearer" who has the right to ask that work be redone __ b) Team votes on each submission's quality X c) Team will ask for individual marking and will identify sections by author __ d) Other:
Team member behaves in an unprofessional manner by being rude or uncooperative	EN OEE WHC FG	a) Team attempts to resolve the issue by airing the problem at team meeting __ b) Team requests meeting with professor to problem-solve __ c) Team ignores behaviour __ d) Team agrees to avoid use of all vocabulary inappropriate to the business setting X

Scenario	Accepted initials	We agree to do the following
Team member assumes or requests that his/her name be signed to a submission but has not participated in production of the deliverable	EN OEE WHC FG	a) Team agrees that this is cheating and is unethical __ b) Friends are friends and should help each other X c) Team will submit with signature but will advise professor who will take action __
There is a dominant team member who is content to make all decisions on the team's behalf leaving some team members feeling like subordinates rather than equal members	EN OEE WHC FG	a) Team will actively solicit consensus on all decisions which affect project direction by asking for each member's decision and vote X b) Team will express subordination feelings and attempt to resolve issue __ c) Other:
Team has a member who refuses to participate in decision making but complains to others that s/he wasn't consulted	EN OEE WHC FG	a) Team forces decision sharing by routinely voting on all issues __ b) Team routinely checks with each other about perceived roles X c) Team discusses the matter at team meeting __

Domain Class Diagram



1.Project Background and Description

Our project goal is to create a blackjack game. Game rules are the following:

1- From 2 to 7 player can play game at a time.
2- The main goal is to beat the dealer without going over 21.
3- Each player starts with two cards, one of the dealer's cards is hidden until the end.
4- Face cards are worth 10. Aces are worth 1 or 11, whichever makes a better hand.
5- To 'Hit' is to ask for another card. To 'Stand' is to hold your total and end your turn.
6- To 'Hit' is to ask for another card. To 'Stand' is to hold your total and end your turn.
7- If you are dealt 21 from the start (Ace & 10), you got a blackjack.
8- Blackjack usually means you win 1.5 the amount of your bet.
9- Dealer will hit until his/her cards total 17 or higher.
10- Doubling is like a hit, only the bet is doubled, and you only get one more card.
11- Split can be done when you have two of the same cards - the pair is split into two hands.
12- Splitting also doubles the bet because each new hand is worth the original bet.
13- You can only double/split on the first move, or first move of a hand created by a split.
14- You cannot play on two aces after they are split.
15- You can double on a hand resulting from a split, tripling or quadrupling you bet.

2.Project Scope

Names of team members:

Elisha: the leader of the group
Omama Eiz Eddin
Furkan Gonul
Wei-Hsiang Change

Roles:

First: choosing a card game.

As for Wei-Hsiang Change his responsibility is to create a public GitHub and upload the Deliverable-1 code to it.

Each one of the team members has to clone the code from our new public GitHub and add his/her signature then push it bake to the same GitHub.

As for Omama her job is to draw a UML class Diagram based on the main code.

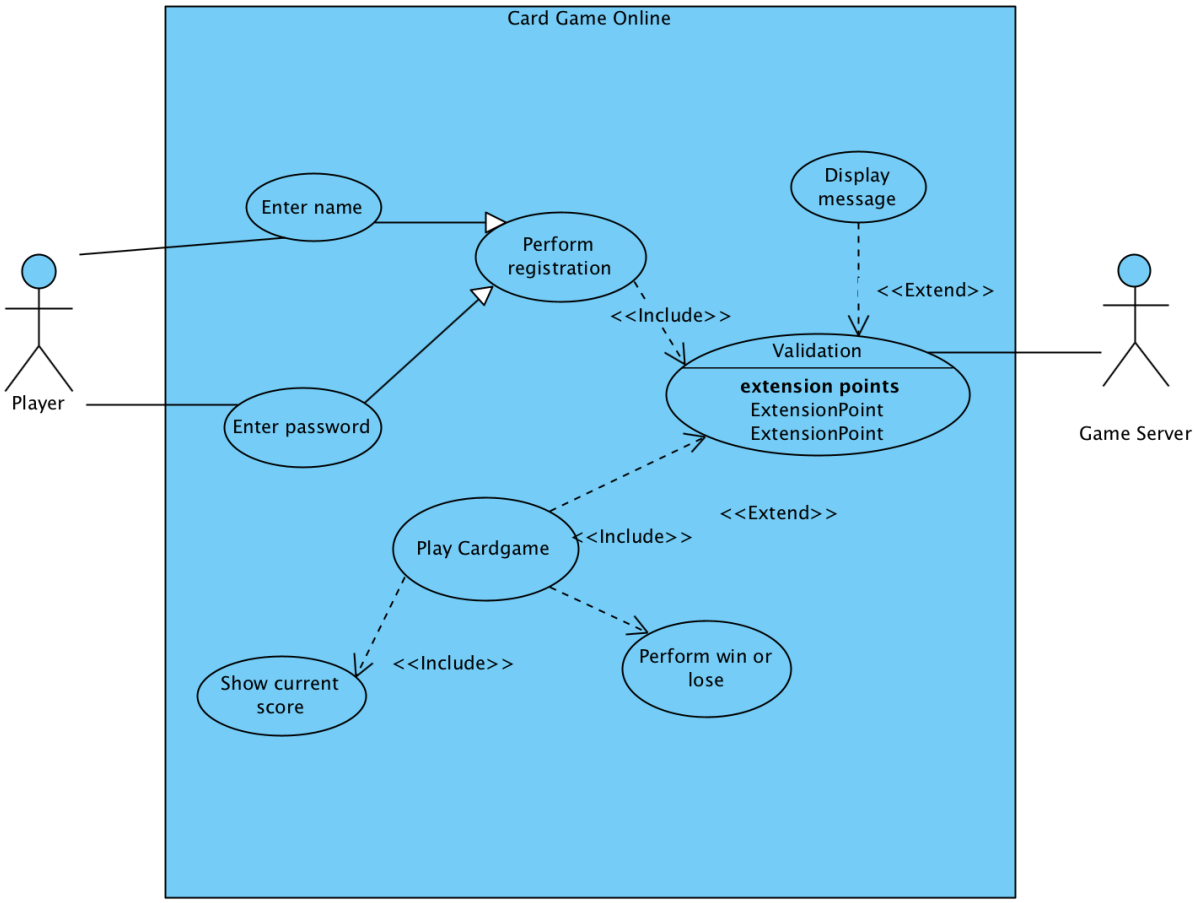
For Elisha and Furkan: They do the first part of Design Document Template. And for the second part Omama and Wei-Hsiang Change will complete it.

The project is basically a card game (Blackjack), and it gives user the ability to register and play with themselves and with the computer.

So, the program will take an input from the users and implement a special technic (based on our code) to interact with the user perfectly then decide the winner and the looser in this game.

Once our project does all these tasks that means it is now completed.

3.High-Level Requirements



4. Implementation Plan

Git Repo : https://github.com/Wei-HsiangChang/Group3_Cardgame

Developer:
Liiish
Omama Eizeddin
robinson4854
Wei-HsiangChang

How to access it:
It is set up for Public, can access without permission.

Expected use:

Developers will clone the code from the repository and then push their code again after they have made changes. The team will be using NetBeans and Github to develop the application and apply changes to the code. The team will use Visual Paradigm to make class diagrams and organize high level requirements. The coding standard used will be OO principles; Maintainable, Extensible, Reusable, and Flexible. Code, Files of separate types will be stored in their own folders, and work will be organized for easy access, viewing and updating.

5.Design Considerations

More Delegation

Dispute different tasks for creating more Class to implement it.
Ex: Creating class like generateCard, ValidatePlayer, Playgame, BlackjackCards…etc

Loose Coupling

Reduce the reliability on some classes, and don't take much work to revise the code to fit different card game.
Ex: Using enum to build Suit and Card value, it is more extendable to create card game.

High Cohesiveness

Using single responsibility principle, don't do too much things on one class. It can be maintainable, and debuggable when dealing with some issue.
Ex: Make a plan to outline the whole system, such as using use case diagram to implement all the function, and focus on specific thing in a class.

Encapsulation

Hiding the implementation, and make the data field and information private, more efficiency to use the class inheritance representing the same behavior. Such as using interfaces and abstract class.