Team Contract

SYST 17796 TEAM PROJECT

Team Name: Polar Bears

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

Please note that if cheating is discovered in a group assignment each member will be charged with a cheating offense regardless of their involvement in the offense. Each member will receive the appropriate sanction based on their individual academic honesty history.

Please ensure that you understand the importance of academic honesty. Each member of the group is responsible to ensure the academic integrity of all of the submitted work, not just their own part. Placing your name on a submission indicates that you take responsibility for its content.

For further information read Academic Honesty Policy on AccessSheridan or visit the faculty office and speak with the Program Support Specialist.

Team Member Names (Please Print)	Signatures	Student ID
Project Leader: Muharrem KAYA		991585597
Derya KAYA		991585609
Mostafa Soroush Zadeh	M.S	991552053

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

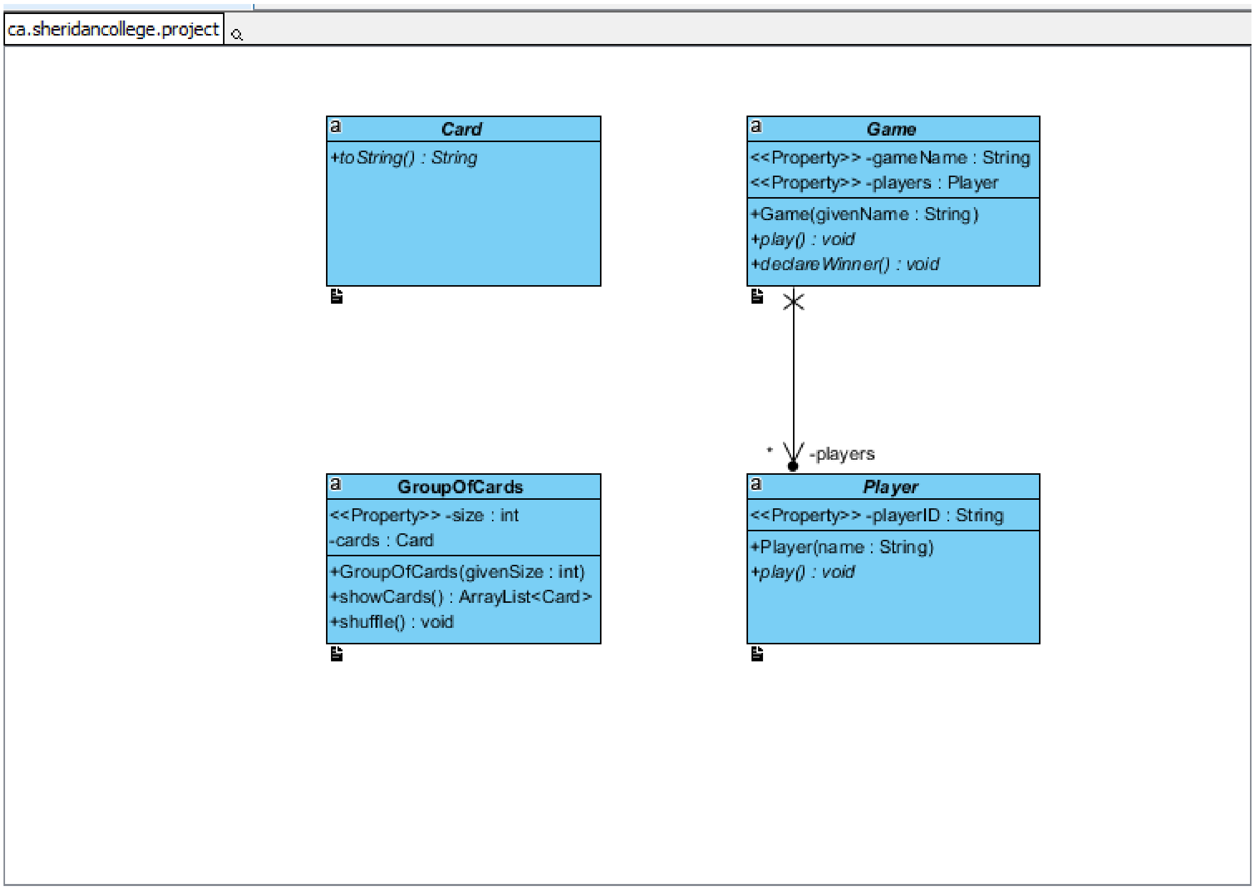
Scenario	Accepted Y/N + initial	We agree to do the following
Team member does not deliver component on time due to severe illness or extreme personal problem	Y + M.S Y MK Y DK	a) Team absorbs workload temporarily Y b) Team seeks advice from professor Y c) Team shifts target date if possible Y d) Other:
Team member cannot deliver component on time due to lack of ability	Y + M.S Y MK Y DK	a) Team reassigns component Y b) Team helps member Y c) Team member must ask professor for reference material Y d) Other:
Team member does not deliver component on time due to lack of effort	Y + M.S Y MK Y DK	a) Team absorbs workload Y b) Team "fires" team member by not permitting his/her name on submission Y c) Other:

Scenario	Accepted Y/N + initial	We agree to do the following
Team member does not attend team meeting	Y + M.S Y MK Y DK	a) Team proceeds without him/her and will assign work to the absent member Y b) Team doesn't proceed and records team member's absence Y c) Team proceeds for that meeting but "fires" member after 3 occurrences Y
A piece of production equipment fails such as a printer, disk drive, or laptop	Y + M.S Y MK Y DK	a) Backup copies will be made and kept in the college Y b) A locker or "share" directory will be used for joint access Y c) A photocopy and duplicate disk of all deliverables will be made Y d) Other:
An unforeseen constraint occurs after the deliverable has been allocated and scheduled (a surprise test or assignment)	Y + M.S Y MK Y DK	a) Team meets and reschedules deliverable Y b) Team will cope with constraint Y c) Other:
Team cannot achieve consensus leaving one member feeling "railroaded", "ignored", or "frustrated" with a decision which affects all parties	Y + M.S Y MK Y DK	a) Team agrees to abide by majority vote N b) Team flips coin Y c) Other:

Scenario	Accepted Y/N + initial	We agree to do the following
Team members do not share expectations for grade desired	Y + M.S Y MK Y D.K	a) Team will elect one person as "standards-bearer" who has the right to ask that work be redone Y b) Team votes on each submission's quality Y c) Team will ask for individual marking and will identify sections by author Y d) Other:
Team member behaves in an unprofessional manner by being rude or uncooperative	Y + M.S Y MK Y DK	a) Team attempts to resolve the issue by airing the problem at team meeting Y b) Team requests meeting with professor to problem-solve Y c) Team ignores behavior N d) Team agrees to avoid use of all vocabulary inappropriate to the business setting Y
Team member assumes or requests that his/her name be signed to a submission but has not participated in production of the deliverable	Y + M.S Y MK Y + D K	a) Team agrees that this is cheating and is unethical Y b) Friends are friends and should help each other N c) Team will submit with signature but will advise professor who will take action Y
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	Y + M.S Y MK Y + DK	a) Team will actively solicit consensus on all decisions which affect project direction by asking for each member's decision and vote Y b) Team will express subordination feelings and attempt to resolve issue N c) Other:

Scenario	Accepted Y/N + initial	We agree to do the following
Team has a member who refuses to participate in decision making but complains to others that s/he wasn't consulted	Y + M.S Y MK Y D.K.	a) Team forces decision sharing by routinely voting on all issues Y b) Team routinely checks with each other about perceived roles Y c) Team discusses the matter at team meeting Y

UML (Class) Diagram


ca.sheridancollege.project
Card
+toString() : String
Game
<<Property>> -gameName : String
<<Property>> -players : Player
+Game(givenName : String)
+play() : void
+declareWinner() : void
*
-players
GroupOfCards
<<Property>> -size : int
-cards : Card
+GroupOfCards(givenSize : int)
+showCards() : ArrayList<Card>
+shuffle() : void
Player
<<Property>> -playerID : String
+Player(name : String)
+play() : void

SYST 17796 DELIVERABLE 1
GO-FISH GAME DESIGN DOCUMENT

OVERVIEW

1. Project Background and Description

The project goal is to design a go-fish game. The game has 52 to cards and the goal is to win the most book of the game, a book is any form of a kind. The game can be plaid between 3-6 player if three player are playing six card will be shuffled to each plyer, if 4-6 player are playing each player will receive five cards. All the player show one card to each other and the player with the lowest rank card will be the dealer, and he/she will be in charge of shuffling the cards counter-clockwise. After shuffling the cards, one of the players starts the game and ask the opponent to give him/her an specific card name for example, ask for the king and the addressed player should hand the card if he/she has it and this will continue and the player can ask for any card as long as the opponent player has the card if the other player doesn't have the named card, in that case the addressed plater will ask the requester to go fish and the player must pick one card of the stack. Whenever the player picks a card of the stack, the card must be shown to everyone than placed in hand, whenever a player gets the fourth card of a book, the player shows all four cards, places them on the table face up in front of everyone and plays again. If a player fails catching the turn passes to the left player. The game is finish when all thirteen books has been won and the player with most book of cards is the winner. During the game each player can take one card if they are out of cards during their turn if there is no card left in the stack they are out of the game. The base code will have enum of cards name and numbers, and arrayList in order to access the cards.

2. Project Scope

Roles of each member:

a. Muharrem (Matthew): designing and testing the game and different classes.
b. Derya: implementing the codes.
c. Mostafa (Michael): coding the game.

The game will be played between two computers and one user and whoever makes the most book of cards will win the give, by reaching thirteen books the game will be over and the winner will be chosen. All the process of the go-fish like catch, go-fish, pick from stack, pass to next player will be implemented in the game.

3. High-Level Requirements

a. One player can be chosen as the dealer.
b. Player can have come-back if they are out of cards. (Only if any card left in the stack).
c. Ability of quitting the game if the player is out of cards.

4. Implementation Plan

Git hub URL: https://github.com/mucteb/Deliverable1

Each member must push the code to the git hub every time code has been changed, modified, or adds. The other members will check the changed or added code and give their feedback on it to other members.

5. Design Considerations

The base class will be implemented by paying close attention to cohesion and independency of this class to the other classes. Also, all of the other classes will be implemented independent from the other classes, expect the go-fish class which it will be implemented by the rule of coupling and it will have some dependency to our game class. encapsulation will be implemented to all the fields and data members in all classes. The go-fish class will be implemented game class(delegation).

6. Referances

Go Fish Rules and Gameplay. (n.d.). Retrieved February 5, 2020, from https://www.ducksters.com/games/go_fish_rules.php