Deliverable 1

SYST17796

Amandeep Sidhu

DUE: June 18th, 2019

Group: Austin DeMelo, Francival Fernandez

Table of Contents

Team Contract

SYST17796 TEAM PROJECT

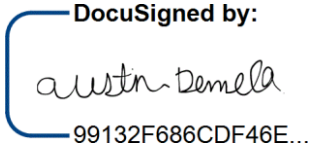
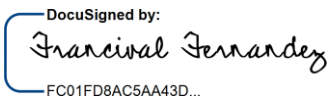
Team Name: Team War

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: Austin DeMelo		Demeloau 991407585
Francival Fernandez		Fefranci 991347868

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	AD FF	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	AD FF	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	AD FF	a) Team absorbs workload _Y_ b) Team "fires" team member by not permitting his/her name on submission _Y_

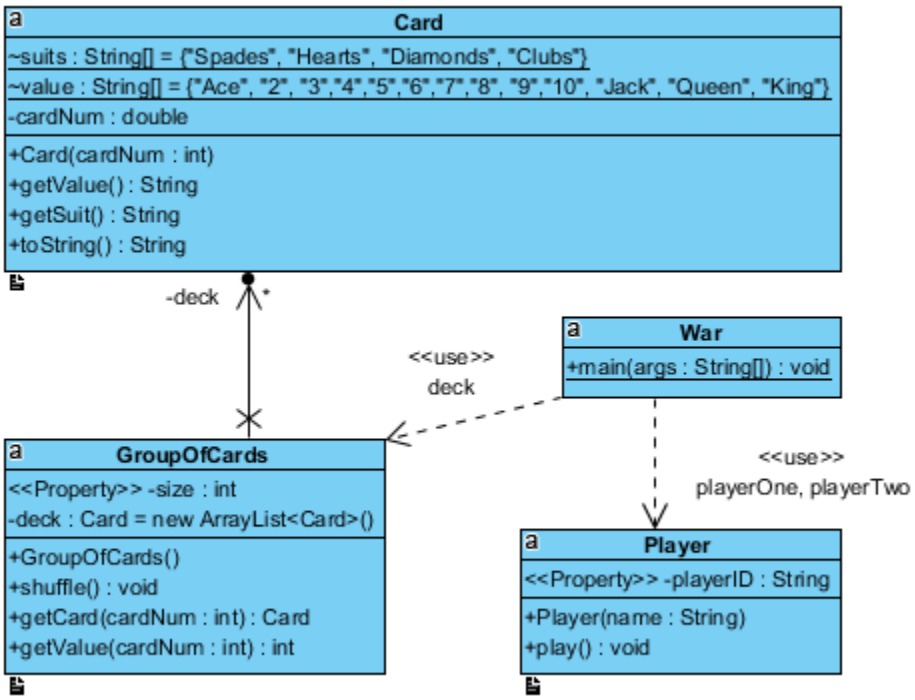
		c) Other:

Scenario	**Accepted Y/N + initial**	**We agree to do the following**
Team member does not attend team meeting	AD FF	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	AD FF	a) Backup copies will be made and kept in the college _N_ 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)	AD FF	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",	AD FF	a) Team agrees to abide by majority vote _Y_

"ignored", or "frustrated" with a decision which affects all parties		b) Team flips coin _N_ c) Other:
Team members do not share expectations for grade desired	AD FF	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 _N_ c) Team will ask for individual marking and will identify sections by author _N_ d) Other:

Scenario	**Accepted Y/N + initial**	**We agree to do the following**
Team member behaves in an unprofessional manner by being rude or uncooperative	AD FF	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	AD FF	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 _N_
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	AD FF	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 _Y_ c) Other:
Team has a member who refuses to participate in decision making but complains to others that s/he wasn't consulted	AD FF	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 Diagram


Card
~suits : String[] = {"Spades", "Hearts", "Diamonds", "Clubs"}
~value : String[] = {"Ace", "2", "3","4","5","6","7","8", "9","10", "Jack", "Queen", "King"}
-cardNum : double
+Card(cardNum : int)
+getValue() : String
+getSuit() : String
+toString() : String
-deck
*
War
+main(args : String[]) : void
<<use>>
deck
GroupOfCards
<<Property>> -size : int
-deck : Card = new ArrayList<Card>()
+GroupOfCards()
+shuffle() : void
+getCard(cardNum : int) : Card
+getValue(cardNum : int) : int
<<use>>
playerOne, playerTwo
Player
<<Property>> -playerID : String
+Player(name : String)
+play() : void

Design Document Template

Background

Our goal with this project is to create a functioning card game that will be played within the chosen IDE and results will be shown in the console. The card game we have chosen to create is War. The base code for this game is written in the Java language. The project is divided into four classes: Card, War, GroupOfCards and Player. We try to follow strict naming conventions and programming principles in order to make our code as easy to follow and understand as possible.

Game Description

We have chosen to create the game of War. The rules of this game are simple, the deck is divided evenly between two players (26 cards each). Each player places their deck face down in front of them. Each player then turns their top card over at the same time, the player with the higher card gets to keep the two cards in a pile (their points). If both pulled cards are the same rank, it is War. Each player pulls out on card face down and one card face up, the player with the higher rank takes all cards (6 points). If the cards are again the same rank the players will repeat the previous action. The game ends when all of the cards have been dealt.

Project Scope

Our group members are Austin DeMelo and Francival Fernandez. Austin is the team leader and his main roles are design orientation and creating a project overview. Francival will be doing the bulk of the code work for the War card game. The interface of the game will be text based. The players will be able to navigate the menu and play the game by typing an the appropriate number for their choice. We will know when the project is complete when we are able to successfully play the War game within multiple trial attempts. We must consider all possible exceptions and possible errors.

High-Level Requirements

The System we are improving must include:

- Ability for anyone with the appropriate files to access and play the game
- Ability for the game to effectively simulate the card game “War”
- Ability for the game to display a win or loss of a round
- Ability for the game to track the deck size of each player
- Ability for game to display the score at the appropriate times
- Ability for the player to replay the game as many times as they would like
- Ability for each player to register with the game
- Ability for each player to change their name before and after games

Implementation Plan

GitHub URL: https://github.com/franfernan/Deliverable1

Each developer is expected to provide their work at the end of each work session. Work sessions are decided amongst the two members of the team, these include times in which both members are available to work. The workspace will be organized in such a way as to dived code from text style documents. This will help to overall keep the project from getting cluttered and will make it easier for the members to find what they are looking for.

We follow consistent coding standards throughout our project. This not only reduces clutter, but it also keeps documents organized. Organizing is one of the largest concepts we strive to achieve, it makes navigating code immensely easy and will reduce the overall work time for the project. Concepts that help us organize our work include: Java naming conventions, DRY (don't repeat yourself), single responsibility principle. Just to name a few. The tools we are expected to implement into our project include: Visual Paradigm, NetBeans, GitHub, Microsoft Word.

Design Considerations

For encapsulation, the variables in the Card and GroupOfCards classes were set to private if possible. Examples of this is the cardNum double variable in Cards and the deck ArrayList in GroupOfCards.

For delegation, the suits/values of the individual cards were delegated to the Card class and the order of cards in the deck was delegated to GroupOfCards.

For flexibility/maintainability, the names of the variables were descriptive enough for anyone to know its function at a glance, and the classes are modular. The Cards class can be used for a card game other than War.