SYST 17796 – FUNDAMENTALS OF SOFTWARE DESIGN AND DEVELOPMENT

DELIVERABLE 1

SUBMITTED BY TEAM 21

TEAM MEMBERS:

- Samarth Goyal	(Student ID: 991658703)
- Anela Ysabelle Navaro	(Student ID: 991674379)
- Sahaj Chawla	(Student ID: 991656276)

TABLE OF CONTENTS

SYST 17796 DELIVERABLE 1
DESIGN DOCUMENT TEMPLATE

OVERVIEW

1. Project Background and Description

The goal and final vision of this project is to create a fully functional clone of the famous card game Blackjack in Java, while adhering to the main pillars of Object-oriented programming. The game will be played using the standard 52-card pack, and the object of the game is for each participant to attempt to beat the dealer by getting a count as close as 21 as possible, without going over 21. At the beginning of the game, the user will place their bet, and the program will then generate two random cards for both the dealer and the player. The user is then given the option to either *hit or stand*. If the user chooses to *hit* the program will give another random card to the player, if the user chooses to *stand* the program will not generate another random card and move on to the dealer. This cycle continues until either the player or the dealer reaches 21 or closest to 21. If the player wins, the player will receive a reward twice the bet amount, however if the dealer wins, the player loses all the money.

The project's starter code contains the following four classes:

- Player Class: This class models each player in the game, and it contains a constructor allowing players to set their IDs.
- Game Class: This class contains an Array List of the players.
- Card Class: This class acts as the base class for all the cards used in the game.
- GroupOfCards Class: This class contains an Array List of the cards.

2. Project Scope

Our team a.k.a. Team 21 comprises of three team members, and we have split responsibilities accordingly:

- Samarth Goyal: In charge of maintaining code
- Anela Ysabelle Navaro: In charge of setting deadlines and goals
- Sahaj Chawla: In charge of keeping things on track

The project's end goal is to create a terminal-based game which comprises of two opponents, *a player* and *a dealer*. The project needs to abide by the rules of the famous card game Blackjack. The project will be considered completed when the program is able to interact with the user and the user is able to understand the rules and steps of the game. Lastly, the game will end with notifying the user of their win or loss.

3. High-Level Requirements

The new system must include the following:

- Ability for each player to register with the game
- Ability for the game to communicate a win or loss
- Ability for players to know their status (score) at all times
- Ability for players to place bets
- Ability for players to get and see their random cards
- Ability for players to either *hit* or *stand*

- Ability for players to see their total out of 21 or how much more they need
- Ability for players to decide whether Ace is equivalent to 1 or 11
- Ability for players to double/increase/split their bets

4. Implementation Plan

The GitHub link for the project's repository is: *https://github.com/Samarth-Goyal/BlackJack-Team21*

Members of Team21 have concluded to check the code repository every third day at the least if not more often. Each member is responsible to notify the rest of the team before making any changes to the code repository. In respect to the file system, we have decided to have two separate sections, one for the code and the second for other submission files such as UML diagrams or documents.

These are the standards and tools we intend to follow and use:

- Standards:
    - Main principles of Object-oriented design and programming
    - Rules set under Sheridan's Academic Integrity Policy
- Tools:
    - JDK 14
    - JUnit
    - Visual Studio Code
    - GitHub
    - NetBeans
    - Visual Paradigm
    - Discord
    - Microsoft Suite

5. Design Considerations

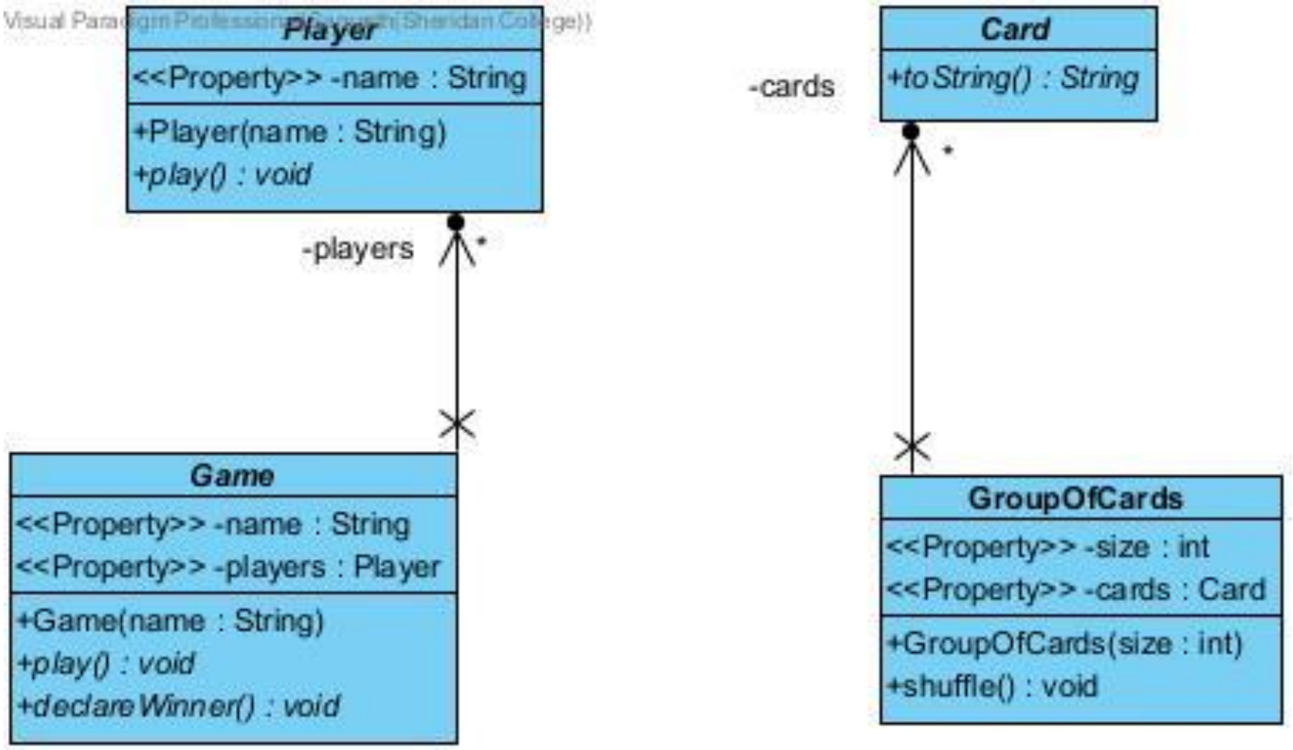
- Encapsulation
    - Data variables of classes are declared as private
    - Data variables' values can be modified or viewed through the provided public setter and getter methods
- Delegation
    - The project's starter code has not implemented inheritance per se but has used delegation as an alternative.
    - For example, all classes serve a specific purpose - class GroupOfCards is delegated the card properties from class Card, which are then stored in an array list.
- Flexibility/Maintainability
    - The project's starter code has used abstract classes in order to share code among several closely related classes.
    - The code also offers modularity, because each class is a separate namespace which helps in organizing the code.

6. References

- Collin, J. (2022). How To Play Blackjack – Blackjack Apprenticeship. https://www.blackjackapprenticeship.com/how-to-play-blackjack/

SYST 17796 DELIVERABLE 1
UML CLASS DIAGRAM

1. UML Class Diagram depicting the starter code:



SYST 17796 DELIVERABLE 1
TEAM CONTRACT

Team Contract

SYST 17796 TEAM PROJECT

Team Name: ___ Team 21______________________________

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.

Team Member Names (Please Print)	Signatures	Student ID
Project Leader: Sahaj Chawla		991656276
Anela Navarro		991674379
Samarth Goyal		991658703

For further information read Academic Integrity Policy on AccessSheridan.

By signing this contract, we acknowledge having read the Sheridan Academic Integrity 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.
- Notifying the professor of any issues in a timely manner so that corrective measures can be taken.
- Any other duties as deemed necessary for project completion.

What we will do if . . .

Scenario	**Accepted initials**	**We agree to do the following (Put an X corresponding to your choice in each box)**
Team member does not regularly attend team meetings and/or does not respond to communications in a timely manner.	S.C. A.N. S.G.	Project leader emails the student citing the concerns and cc's the professor so they are aware of the situation at the very onset _X_ (**Mandatory**). a) __ In addition to above, the leader/team will (specify):
Team member does not deliver component on time due to severe illness or extreme personal problem.	S.C. A.N. S.G.	a) Team absorbs workload temporarily __ b) Team seeks advice from professor __ c) Team shifts target date if possible __ d) __ Other (specify):
Team member has difficulty delivering component on time due to lack of understanding or ability.	S.C. A.N. S.G.	a) Team reassigns component __ b) Team helps member __ c) Team member must ask professor for help __ d) __ Other (specify):
Team member does not deliver component on time due to lack of effort.	S.C. A.N. S.G.	a) Team absorbs workload __ b) Team member(s) ask professor to request a Participation Form from <u>all</u> team members. This *may* result in individualized grades being awarded for a deliverable __ c) Both a. and b. above __ d) __ Other (specify):
Team cannot achieve consensus leaving one or more member(s) feeling that their voice(s) is/are not being heard in a decision which affects everyone.	S.C. A.N. S.G.	a) Team agrees to abide by majority vote __ b) Team seeks advice from the professor __ c) __ Other (specify):

Team members do not share expectations for the quality of work on a particular deliverable.	S.C. A.N. S.G.	a) Team members will draw on each other's strengths to help bring the quality of the deliverable to a minimal acceptable level __ b) Team votes on each submission's quality __ c) Team member(s) ask professor to request a Participation Form from all team members, which may result in individualized grades being awarded for a deliverable __ d) __ Other (specify):
Team member behaves in an unprofessional manner, e.g. being rude, uncooperative and/or making one or more member(s) feel uncomfortable.	S.C. A.N. S.G.	a) Team agrees to avoid use of all vocabulary inappropriate to a business/college setting __ b) Team attempts to resolve the issue by airing the problem at a team meeting __ c) Team requests a meeting with the professor to discuss further __ d) __ Other (specify):
There is a dominant team member who insists on making all decisions on the team's behalf leaving some team members feeling like subordinates rather than equal members	S.C. A.N. S.G.	a) Team will actively solicit consensus on all decisions which affect project direction by asking for each member's decision and vote __ b) Team will express subordination feelings and attempt to resolve issue __ c) Team seeks advice from the professor __ d) __ Other (specify):
Team has a member who refuses to participate in decision making but complains to others that s/he wasn't consulted	S.C. A.N. S.G.	a) Team forces decision sharing by routinely voting on all issues __ b) Team routinely checks with each other about perceived roles __ c) Team discusses the matter at team meeting __