**Introduction (10 pts)**

**o Explain what the software project about and what are its goals: 2 pts**

Software project is about a simple program that implements the soccer game which combines the following topics learned in this course: UML, different patterns like structural, creational, behavioral, and testing. The goal is to get familiar with and hands on experience on this task. Creating buttons that function properly like the game should be paused initially until the user resumes or enters the key R. Likewise, Q- quitting the game, P– pause the game, R- resume the game, Spacebar- to shoot an arrow key to move the striker. Finally, the score should be displayed on the top left part of the screen.

**o Explain the challenges associated to the software project: 2 pts**

A lot of running time errors due to a few spelling errors. Some program requirements were not fulfilled so I had to redo the work. It was confusing when there was no iterator other than linear search in PlayerCollection and the program used to work fine but we were asked to implement a separate PlayerCollectionIterator class. Some other difficulties were to keep track of the goals scored and the saves by the goalkeeper.

**o Explain the concepts (e.g., OOD, OOD principles, design patterns) you will use to carry out the software project: 4pts**

Object-oriented design (OOD) is usually used to increase the accuracy of the program and reduce its development time. Programs that are created using OOD are easier to write and modify.

Principles of object-oriented designs:

Abstraction: Hiding unnecessary implementation details of one class from another class, only relevant methods that are needed to interact are exposed.

Encapsulation: Objects of two classes interact with each other.

Polymorphism: Methods need to function according to the parameter input or object using it.

Inheritance: Parent and child have very similar states and methods, where child class can reuse the parent class state and method without altering them.

Design patterns are useful when we analyse software development life cycle, they ease the analysis and requirement phase by providing information based on hands-on experiences. We are going to use three design patterns in this project: Creational, Structural and Behavioral design patterns. We use it because they provide transparency to the design of an application. Although they don’t provide an absolute solution to the problem but provide clarity and possibility of building a better version of it.

**o Explain how you are going to structure you report accordingly: 2pts**

| **Reports** |
| --- |
| PART I (introduction): up to 1 page |
| Part II (design): up to 3 pages |
| Part III (implementation): up to 3 pages |
| Part IV (conclusion): up to 2 pages |

**PART IV: Conclusion (10 pts)**

* **What went well in the software project? 2 pts**

Overall, the project was fine and not very difficult to work as a lot of the parts of implementation were already provided. It was the first group project so I had a feeling of how we are supposed to collaborate in future as a software developer at a company. Everyone made efforts to complete their part on the scheduled time and fulfilled the project requirements. The professor and TAs helped us with additional support and resources when we were falling behind and stuck at a problem.

* **What went wrong in the software project? 2 pts**

There were lots of defects and errors that occurred during the testing of the code. Some part of the implementation was not necessary, but we still had to as instructed in the pdf provided. Some solutions did not have the requirements that were asked which led to some confusion and misunderstandings. After a working model too, there were changes to be made which made us rework on the project and caused delay and we had to work on the day of submission too.

* **What have you learned from the software project? 2 pts**

How to effectively use the principles of object-oriented design and the design patterns as it helps to keep your code organized. Also, to work in a group as this was one of the courses where we had a collaborative assignment project.

* **What are the advantages and drawbacks of completing the lab in a group? 2 pts**

Advantages are you get to learn how to work in a group, everyone has their own way to find a solution to the problem with a wide range of ideas. Drawbacks are the time management or conflicts. Not everyone has a mutual time to work on the assignment, everyone does it at their own time which is acceptable but a little frustrating sometimes.

* **What are your top three recommendations to ease the completion of the software project? 1pt**

1. Read the instructions carefully and following them will get an absolute solution.
2. Start working on it from the day it was released that eases up the pile of work, that is efficient rather than getting things done just before the submissions.
3. Create a way to keep a track of requirements.
4. Prove the software project conditions.

* **Add a paragraph (or a table) to indicate the different tasks of the work that were assigned to each group member, and the portion of the work completed by each group member. Also indicate if each group member was collaborative. 1 pt**

The tasks that were assigned to each group member are described briefly, and the green color represents the task completed.

| **Aniket Pethani**  **(Student ID - 216927543)** | **Mustafa Ahmad 0%**  **(Student ID - )** | **Ahmad Ali**  **(Student ID - 214878904 )** | **Gurkirat Bamrah**  **(Student ID - 215082704)** |
| --- | --- | --- | --- |
| PART 1: Introduction (reports) | Part 2: design & UML | Part 3: Implementation of JUnit | Additional classes mentioned the pdf on page 14. |
| PART 4: Conclusion (reports) | 0% | Running the JUnit | Implementation of mvc |
| Video: Launch and run the code |  |  |  |