1. **USE CASE:**

**Matching Game**

A matching game starts with players selecting a character group, such as fruits. They then embark on a timed challenge to strategically swap the positions of characters within a two-dimensional matrix. The aim is to align three or more identical characters in a row, column, or diagonally, earning points for each successful match. To add some challenges, there is a countdown timer ticking away. However, the fun part is that the player competes with the computer in a friendly competition.

1. **INTRODUCTION:**

This project is about making a game with Python for children. In this simple game, children can find similar objects and put them beside each other. If it is, they will get a score. They can play this game with friends and enjoy it. The challenge of this game is getting a higher score in a limited time and winning the game. The player with the higher score will be the winner of the game.

1. **System Overview**

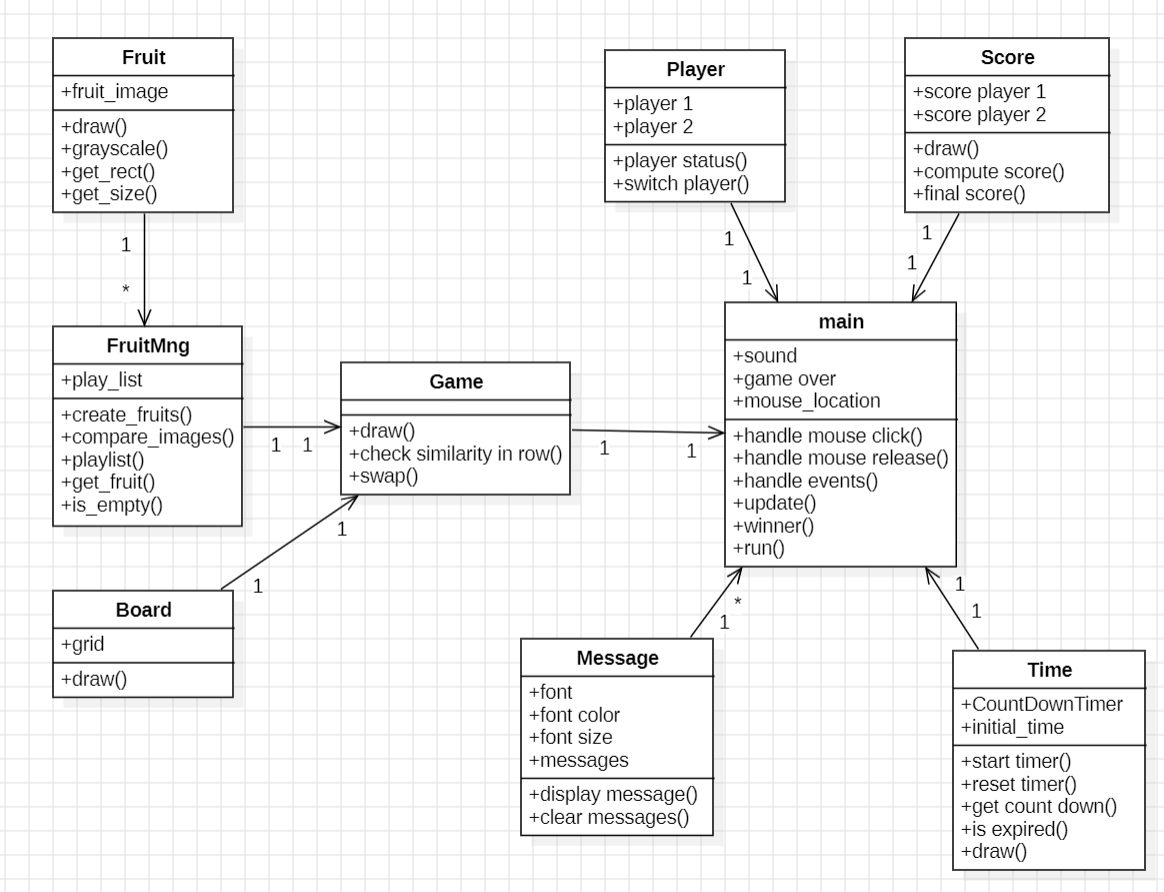
**What will be built and the scope of the project:**

List of requirements: according to the deadline of the project, the list of requirements and what will be built, is determined in the Engineering assessment column.

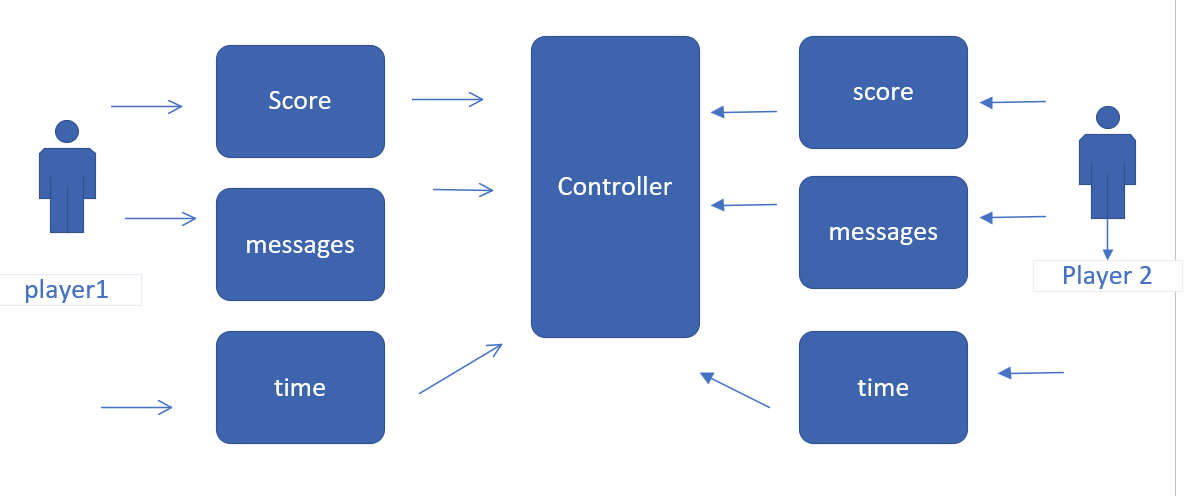
|  |  |  |  |
| --- | --- | --- | --- |
| **Code** | **Requirement** | **Priority** | **Engineering assessment** |
| **001** | players selecting a character group, such as fruits | medium | Not Satisfied |
| **002** | time challenging | high | Satisfied |
| **003** | swap the positions of characters within a two-dimensional matrix. | High | Satisfied |
| **004** | align three or more identical characters in a row, column, or diagonally | High | Satisfied, just for row |
| **005** | earning points for each successful match. | High | Satisfied |
| **006** | countdown timer ticking away | High | Satisfied |
| **007** | player competes with the computer in a friendly competition. | medium | Not Satisfied |
| **008** | two-dimensional matrix | High | Satisfied |

1. **Architecture and System Design**

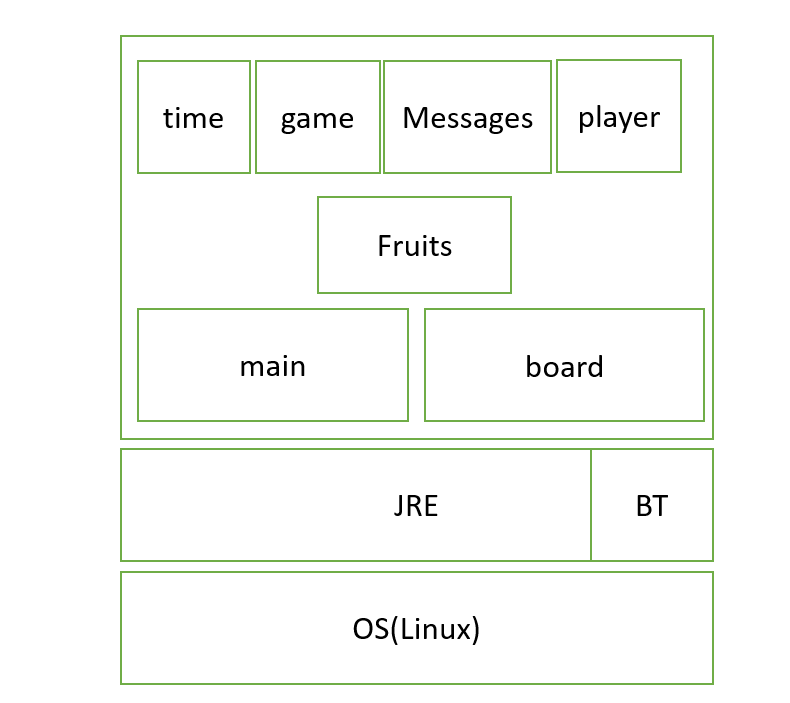
**UML**:



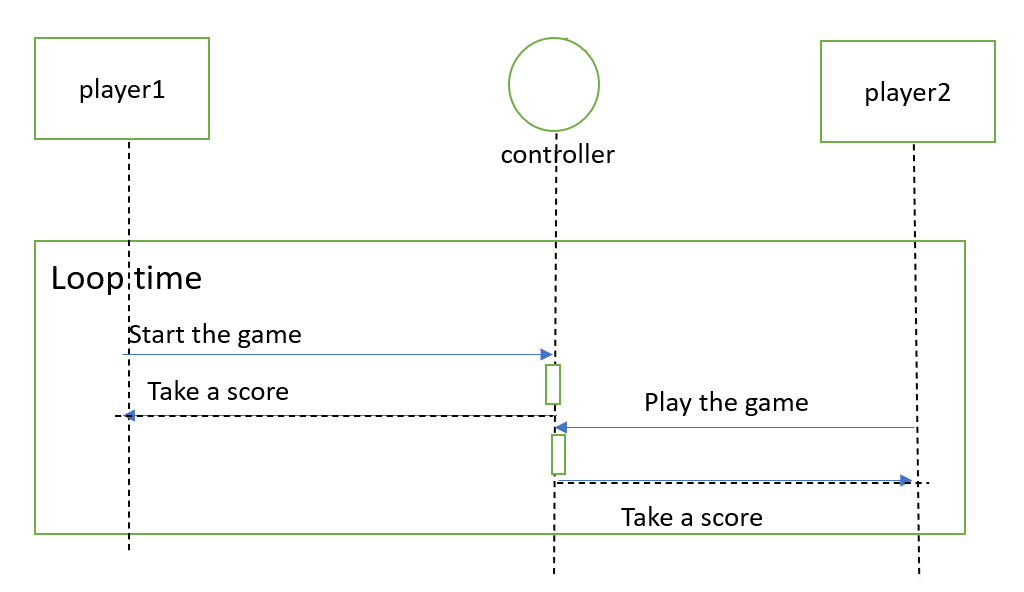
**ALD:**

****

**AFD:**



**sequence diagram:**



1. **Subsystem and Feature Design**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Module mapping** | |
| swap the positions of characters | Class of Game | Drawing the game on the main class |
| align three or more identical characters in a row | Class of Game |
| earning points for each successful match. | Class of Score |
| countdown timer ticking away | Class of Time |
| two-dimensional matrix | Class of Board |