Yazan Alasad Masoud,
Aleksey Vanshin

TDD

1.Features

- 2D
- Environment including
  - 2D Objects
  - Player
  - Obstacles
- Storytelling
- Interactive objects
  - Obstacles
  - Enemys
  - Plattforms
- Realistic physics

2.Choise of Game Engine

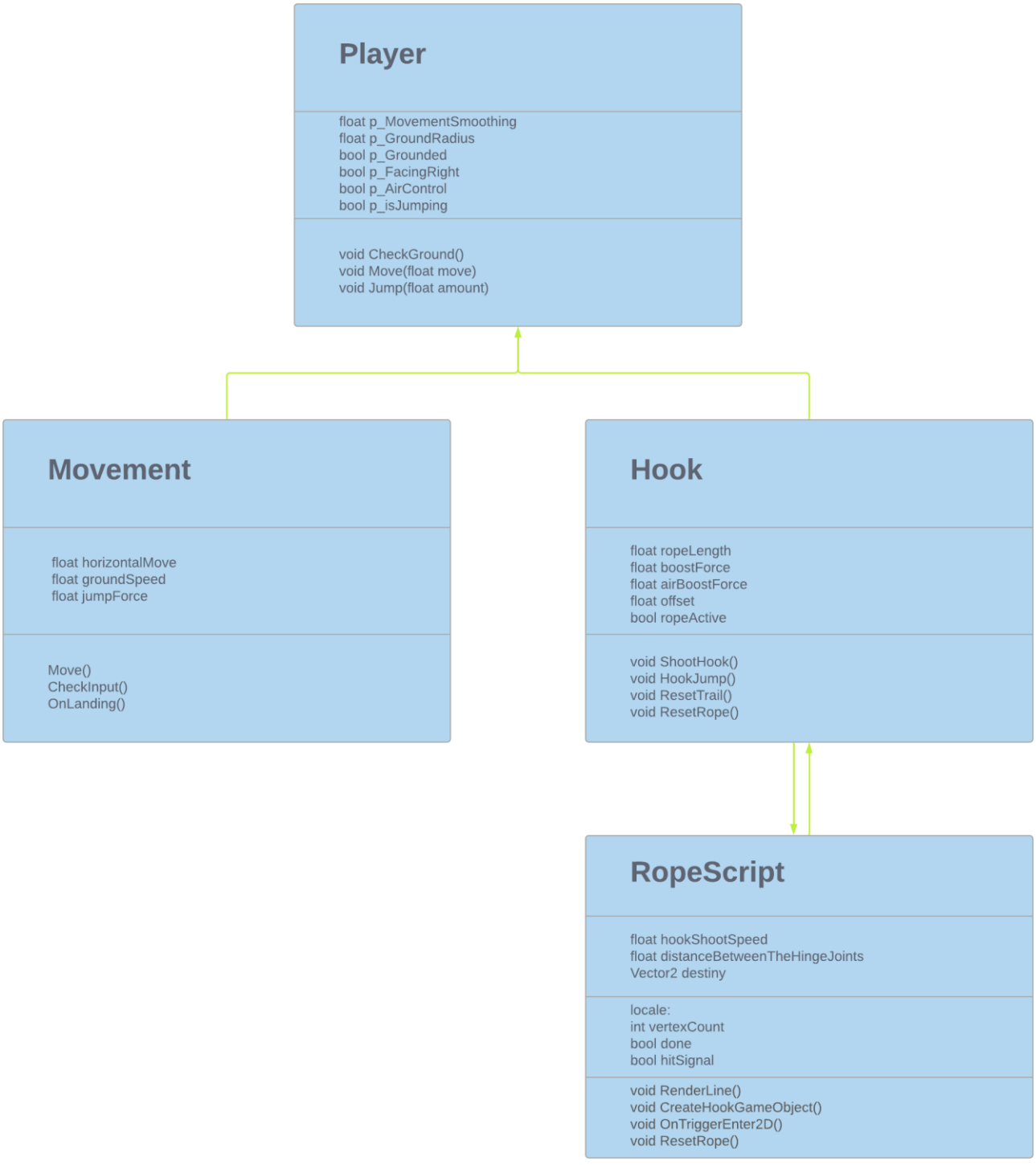
The game engine that has been chosen is Unity.

- Physics support with physics assets
- Great for creating nice looking 2D scenes
- Rendering

The reason that Unity engine was shosen is because this engine is the most common useful engine in which all the teammembers the most experience had.

Yazan Alasad Masoud,
Aleksey Vanshin

3.Code Architecture



Yazan Alasad Masoud,
Aleksey Vanshin

4.Naming Conventions

- Code:

Private	camelCase
Public	PascalCase
Lists	PascalCase
Const	UPPERCASE
Regions	camelCase
Classes	PascalCase
References	camelCase
Structs	PascalCase

- Files:

Single Instance	PascalCase
Multiple Instance	PascalCase_X

- Folders will be written in “PascalCase”

Yazan Alasad Masoud,
Aleksey Vanshin

4.Unity Build

- SceneName:
  - Menu(Index[0])
  - IntroCutscene(Index[1])
  - Game(Index[2])
  - EndCutscene(Index[3])

Yazan Alasad Masoud,
Aleksey Vanshin

5. Collision Detection, Physics & Interaction

- **Collision detection:**
    - Player hit an obstacle
    - Player hook
    - Objects falling
- **Physics:**
    - Items/objects fall over realistically
    - Items/objects deform realistically
    - Friction and gravity
    - Objects and player must react to player input and his decisions(push/pull objects)

- **Interaction:**
    - Interaction with objects/items:
        - Pushing/Pulling
    - Environment interaction:
        - POV cutscenes
        - Sound feedback
    - Layer need to be set to “Grappable” for proper interaction between hook and object

6.Audio

- Internal audio sound producing
- Internal visual effects producing

Yazan Alasad Masoud,
Aleksey Vanshin