Project plan

Scope and Schedule

Our task is to make a clone of the popular game Angry Birds. In the original game you throw birds at green pigs and try to destroy them all. We thought we could spice things up a notch so we will use some other type of throwables and enemies. This will make the game a little bit more personal.

The project will done over a 6 week long period in stages. The first stage will be to get one “throwable” and one level to work and then additionally add more features and levels. The schedule would look something like this.

Nov 4 - Nov 10	Base classes implemented
Nov 18 - Nov 22	Have a working base level with one “throwable”
~Dec 1	Implementing more “throwables” and levels
~Dec 8	Finalizing graphics and physics
Dec 12	Final commit to git

We will have two base classes “Creatures” and “Map” and several subclasses. This will be more explained in the class hierarchy. Things will surely change when the project is on its way.

Roles

The Angry Birds project requires to create a game which will be based on the programming part and the user experience part. For this reason we have considered to include the following roles in the development of our game:

- **Project Manager**
  The Project Manager role (PM) is in charge of the overseeing of the project by knowing the project time management, quality management, risk management, scope management and stakeholder management. In this sense, the PM communicates effectively with all the stakeholders involved in the project, like the internal stakeholders (team members) and the external stakeholders, e.g. our project advisor or other students. In addition, during the

project execution, this role will also be in charge of performing the change management properly, since during the project lifecycle there will be several changes on our software project, the PM along with all the team members should assess the changes, specially to evaluate their impact and decide its incorporation or not within the project.

- **Back-end developer**
  In this role, the back-end developer will be in charge of building the deliverables defined in our scope and design architecture by constructing the software parts based on the requirements and the make sure the features we are defining in our project plan are met according to the specifications. For our particular project, the developer will be the person who is in charge of implementing the class hierarchy, providing physics to the objects in our game and implement the logic, for instance. In addition, the developer role must assess constantly the risk when creating the deliverables and inform to the PM and team members about any impact on the project schedule, quality and scope, for this reason this role is also an active stakeholder in the change management process of our project.

- **UI/UX designer**
  Our UI/UX designer role will be specially in charge of making our game an easy-to-use software to achieve an engagement from the user when they play. The designer will be designing the graphics and functional features of the project by designing the user interface elements like menus, combos, tabs and any other components that will make the game a good and easy user experience. Also, the designer role will be assessing the requirements and features to check for any functionality to be modified, for this reason, this role as with the others will be deeply connected with the change management process of the project.

In our Angry Birds project, we have divided the roles as follows:

Member	**Role**
Gabriel	Project Manager
Kim	UI/UX (Back-end)
Simon	UI/UX (Back-end)
Wesam	UI/UX (Back-end)

It is important to notice that all the team members will be creating the deliverables of the project and participating in different roles, for this reason the Project Manager role will also be acting as Back-end developer for instance.

Features

Basic Features

Like every Angry Bird type of game it will have the basic features. It is going to have some sort of basic graphic where the player can see the map, the objects that they will shoot and the targets they have to destroy. The game will also have a couple of different objects with some special features that the player can use at its advantage, for example explosion or bounciness. The game is controlled by computer mouse and works by pulling the object backwards while holding in the mouse button scope at the direction you want to shoot and release the button to let the object fly away. The game is also incorporating a couple of different levels with increasing difficulty to complete. It will show how many shots you have left and the score you have made in the round. The different maps are loaded from files and have physics defined for them. The game should collect points when you play the game, you get points for enemies killed, other objects destroyed and how few objects have thrown. It also shows the player how many objects are left and enemies left.

High score

The game is going to have a high score features were the player can see the highscore on the current level their playing, maybe in the top right corner. The plan at the moment is that in the start of the game you type in your user name and after every game it checks if you improved the high score, if that's the case it overwrite the current high score and shows your name and score instead.

Stars

In the end of the game, if you accomplished to kill all enemies, you get a rating in stars how well you performed. The stars you get are counted out from the points you collected and amount of object you used to complete the level. The player get graded from one to three stars and the stars are saved to the map file under your username.

Sound

The game should make some various sound like for example when you hit your enemy or when you are shooting away your objects.

Game Modes

There will also be some different type of game modes. Of course there is the original game mod where you shoot objects and have to kill enemies. But also a game mode where you have to complete the levels as fast as you can and and get rated after the time the player takes to complete the tasks.

Better graphics

We will use different kind of materials which have their own abilities and therefore we will need some kind of visualisation for the different materials. This means different “pictures” for wood, metal, ice etc. We will also implement some kind of nice background and landscape (soil, walls etc.). The throwables will also be real “creatures” not just a circle that is thrown in to the enemies fortress.

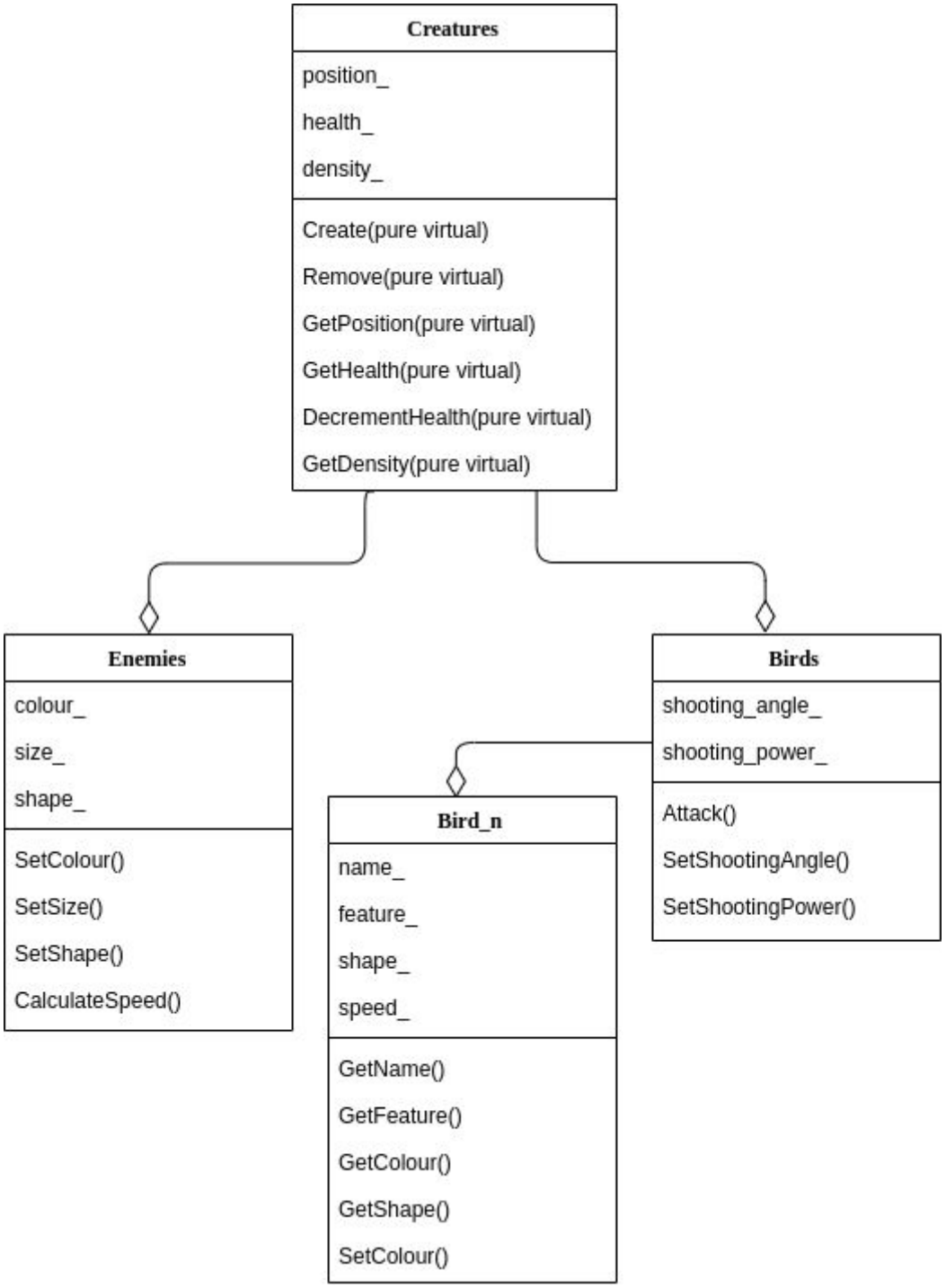
Design and architecture

The game will be running on Linux platform with the following specifications:

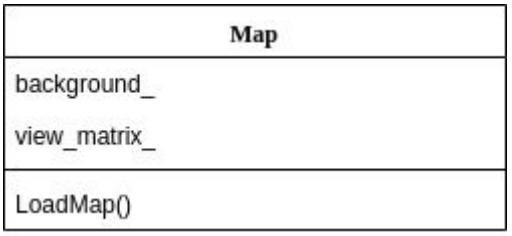
- **OS:** Ubuntu 18.04
- **Physics library:** Box2D v2.3.1
- **Multimedia platform:** SFML v2.5.1/SDL v2.0
- **Programming language:** C++

Class hierarchy

Creatures, Enemies and Birds



Map



Materials