Technical Specification

Analytics Simulation

BEGUE Sébastien

22/04/22

Revision History			
Date	Version	Description	Author
06/02/2022	1.1	Add explanations regarding the implementation and architecture of AI	Carmen BURNY
17/06/2022	1.2	Added explanations regarding the architecture of neural network and test	Carmen BURNY

Introduction

Purpose

This project consists in developing an analytical software for life simulation. This software should allow to perform statistical studies on the evolutionary choices of the species in our world.

Intended audience & pertinent sections

This document should provide information about the project, such as listed features for developers, hardware requirements for users.

Document conventions

This project use the naming and structural convention provided by Microsoft which are described on the following two web pages:
https://docs.microsoft.com/fr-fr/dotnet/csharp/fundamentals/coding-style/identifier-names
https://docs.microsoft.com/fr-fr/dotnet/csharp/fundamentals/coding-style/coding-conventions

Description

Project context

This is an end of year study project of EPITECH M.Sc Pro training. It has been created to match the Green Tech domain of activity and cover multiple core aspect of the M.Sc.Pro training which are the Artificial Intelligence and the Virtuality Realty. The supplier shall deliver a life simulation analysis software tool.

The project has been inspired by different sources such as:

Coding Adventure: Simulating an Ecosystem - YouTube
Simulating Natural Selection - YouTube

Features

- Allow users to launch simulation of life in an environment, for ex. an ecosystem
- Simulations will be displayed in an 3D environment
- Simulations incorporate artificial intelligence within all the defined life beings
- Capable of generating graphics metrics based on simulation result.
- Allow user to customize simulation

Operating environment

The software must run properly (24fps or higher) on Windows 10 x64 with the following minimum requirement:

CPU: Intel i7 8550 1.80GHz

GPU: Intel UHD Graphics 620

RAM: 8GB

Components

Simulation customization

ID REQ_001

- Specie configuration
- Environment configuration

User interface

ID REQ_002

- Simulation window
- Inspector
- Navigation tab

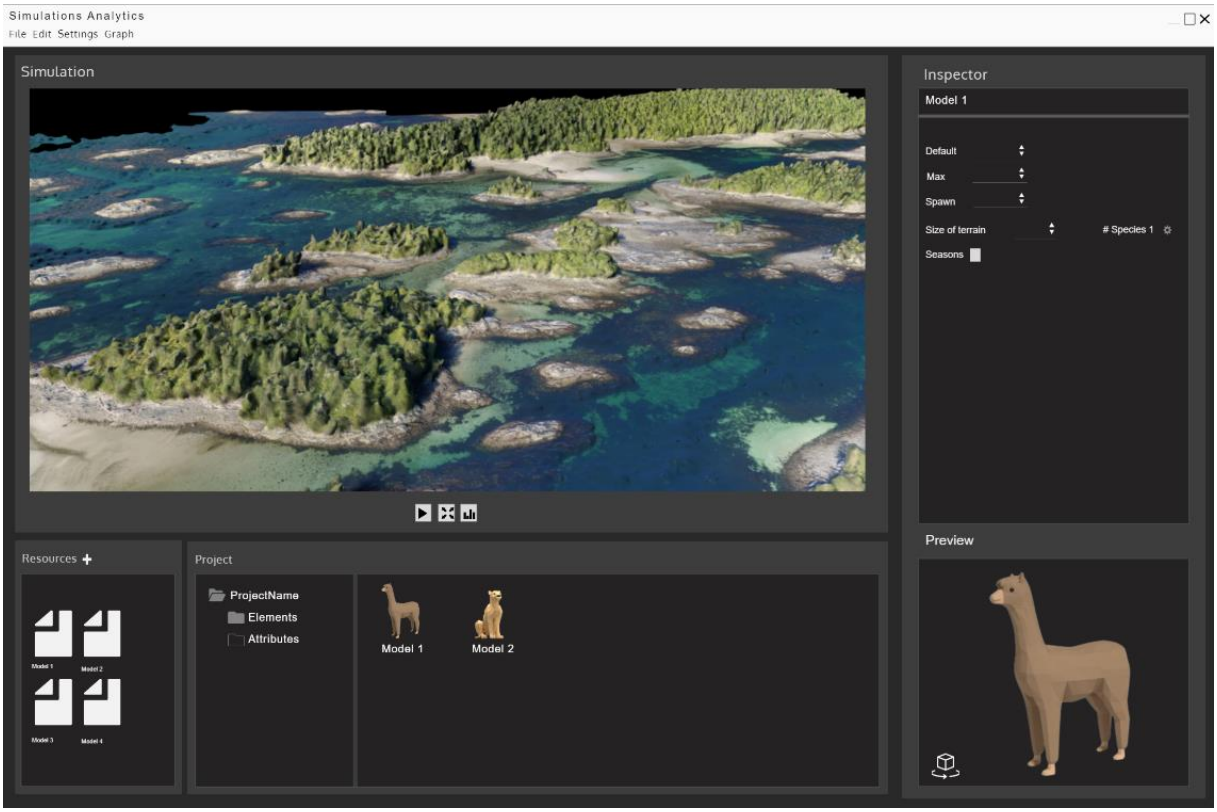


Figure 1 User Interface Design

Models (Animals / Plants / Living Beings)

- Artificial intelligence system

Requirement ID	Description
REQ_001_001	**Must** be able to export specie
REQ_001_002	**Must** be able to import specie
REQ_001_003	**Must** be able to create a new specie
REQ_001_004	**Should** be able to modify specie stats
REQ_001_005	**Must** be able to define the number of animals of each species
REQ_002_001	Simulation window **must** be able to display 3D environment
REQ_002_002	Configuration window **must** let the user configure the simulation.

AI Implementation and Architecture

All beings in the environment will have a pattern of thinking and actions of their own that evolves with time. All behaviors and actions including some movements will be based on output from the calculations of the neural network of the being.

All AI implementation will be extended, in terms of development, from specific classes as the base of having an AI in the beings developed, as per figure shown below.

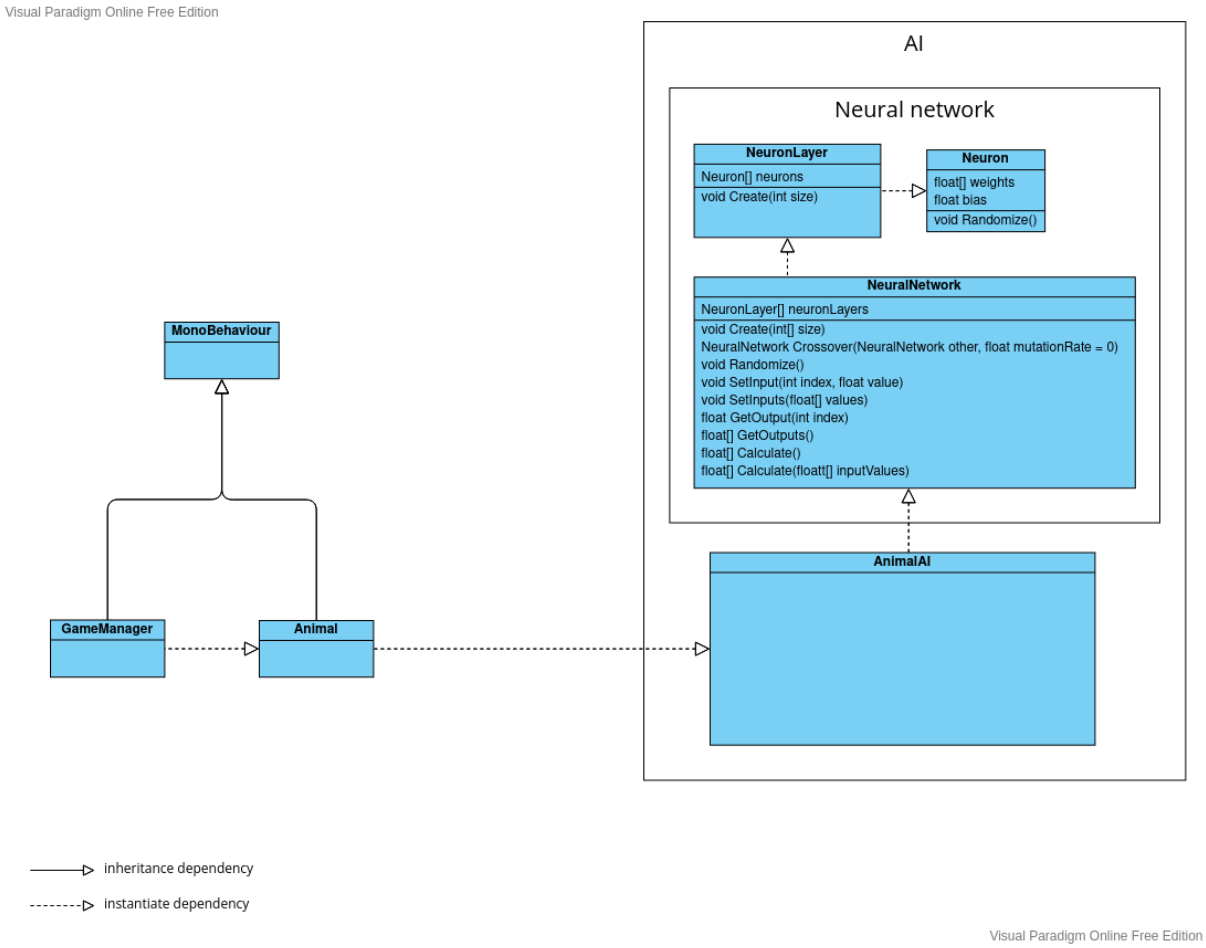
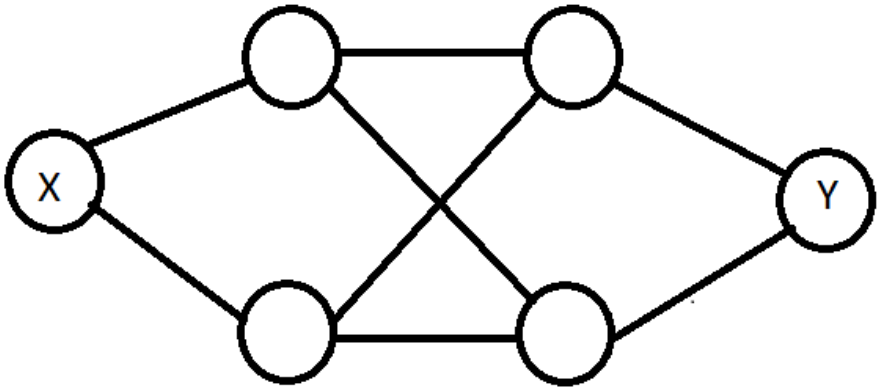


Figure 2 Class diagram depicting AI architecture

The inputs are usually the environmental factors due to an action or state, represented by numbers which will go through calculations through multiplications and addition of weights and bias. Eventually the neural network will generate an output representing the outcome that is of consequence of the actions and environmental factors.

The concept of learning here incorporates the concept of Darwinism, in which a living being will develop through natural selection instead of any objective learning. The structure of neural networks is as per below. The number of neurons in each layer could varied upon taking into consideration the input of user's configuration.



Each and every living being will have a neural network as shown in the figure above implanted in them. Y represents the goal or a state, and X is the input representing action or state. The calculations will be randomized and be executed automatically as the simulation persists. This will change the state of the course of the environment through time, i.e., if an animal is not evolving fast enough or lack of food around them, they'll have a high chance of not surviving in the environment.

Testing

There will be two types of testing implemented which include functional testing, regression testing and integration testing as well as user acceptance testing before finally delivering the software.

Unit Test

As unit aims to test smaller pieces of code, it will be developed separately and on testable cases.

As we create models and functionalities, we will implement the unit test project accordingly, mainly for the integration of different modules. It will be in a separate project that will only be used for testing the methods of the main project.

Functional Test

Any functions will evidently be put through a preliminary testing phase by the developer to ensure that it is achieving what it was developed for.

As the AI was being developed, tests were carried out from the level of bare neural networks to it's implications controlled by the simulation.

Integration & Regression Test

Once the source codes are developed, it will be merged with the already tested codes. This will occur depending on the number of functionalities generated or an ensemble of modules. Once integrated, we will test if the newly developed functions generate any errors and will then note the incidents. This test will be corrected first and then the integration test will be launched again.

If everything went well, we will be testing if other functionalities work as, it should. If error occurs, the same steps of that of the integration tests will be taken until the new functionalities are integrated with no regression occurring.

User Acceptance Test

Test cases in the point of view of users to be tested as the project grows. Each test case will be graded and could have remarks regarding the functionality of the cases.