

Product Plan

04/05/2016



CodeFox

Marco Houtman

Ronald van Driel

Joshua Slik

Matthijs Halvemaan

Lisette Veldkamp

Contents

1. Introduction	3
2. Product	4
2.1. High-Level	4
2.2. Roadmap	5
3. Product Backlog	6
4. Definition of Done	7

1. Introduction

Tygron is a software company focussed on developing the Tygron engine which is used to streamline the planning process and substantially reduces the time and costs involved in urban projects. This is done by building an urban planning project and let users take on their respective roles within the project.

It is not always feasible to have all respective roles within one place which makes it harder to run the game. This can be solved by introducing virtual humans which will take on their respective roles.

In this project we will develop our own scenario. The game that we want to create will revolve around the surroundings of the TU Delft. The five roles in this scenario are the municipality, the TU Delft, DUWO, business facilities and a housing corporation.

The Virtual Human developed by our team will take on the role of a housing corporation. It will focus on finding locations to build houses which are primarily focussed on private housing.

In this document we will talk about how we want our product to be using the MoSCoW method and how we are planning to achieve this. This will be shown in the roadmap. We will also show some user stories in the product backlog to illustrate how our agent should work. Finally we will discuss our Definition of Done, which we will use during each sprint to check if we indeed have a working product.

During this project we will be using the SCRUM method. In this method, weekly sprints are used to create a working product every week. Meaning that after every sprint, everything that needed to be implemented within that given sprint has to be completed and thoroughly tested. We think that SCRUM will help us to manage all of the tasks throughout this project.

2. Product

In this chapter we will discuss the high-level functionalities of our agent and how we are planning to achieve this in the time that we have for the project.

2.1. High-Level

Below we have specified functional requirements for our product, this is done using the MoSCoW method, which means that all requirements are split into the essential things our agent must have, the high priority extra functionality our agent should have, the low priority extra functionality our agent could have, and finally some functionality our agent won't have, due to both technical and time constrictions.

Must have

Our virtual human **must** meet the following requirements:

- Able to create decisions based on environment.
- Able to create decisions based on individual gain.
- Able to interact with other players.
- Able to determine how well it did according to an individual score.
- Able to judge what good and bad decisions are based on all available information.

Should have

Our virtual human **should** have the following features:

- Able to ask other parties to perform actions which are beneficial to the goals of our agent.
- Able to react to propositions from other parties which may or may not be beneficial to the goal of our agent.
- Able to make decisions which have a negative effect on its goals, if it has a great benefit that can be utilized later on.

Could have

Our virtual human **could** have the following features:

- Able to develop advanced strategies based on expectations.

Won't have

Our virtual human **won't** have the following features:

- Able to use and interpret natural languages to interact between players.
- Able to use online datasets to gain additional information.

2.2. Roadmaps

In this section we will illustrate how we think that we are going to achieve our end product. We will have a sprint every week and therefore also a finished product every week, from which we can move forward, until we eventually have created a fully functioning agent.

Sprint 1:

Explore the Tygron engine. Additionally discussions are to be made to determine which role each team gets to simulate.

Sprint 2:

Develop the game environment with all of the virtual humans that will interact with each other. Discuss possible conflicts and create a game containing some conflicts so that it will be challenging.

Game demonstration.

Sprint 3:

Improve game based on feedback from demo. Develop an agent in GOAL which can use basic game actions: Buying land, Selling land, Create and demolish buildings.

Sprint 4:

The agent should be able to reach his indicator targets.

Agent demonstration.

Sprint 5:

Improve the agent according to the feedback from the demonstration. The agent should be able to track and communicate with other agents.

Sprint 6:

Improve strategies of the virtual agent. The agent should be able to have intelligent interactions with the environment.

Sprint 7:

Validate whether the agent has all of the product requirements we set. Bugfixing and implement possible additional features.

Agent demonstration.

Sprint 8:

Finalize product main functionalities. Then focus on minor quality improvements and fixes of minor issues.

Final demonstration.

3. Product backlog

In this chapter we will illustrate how our agent should function in the game. We will do this by using User Stories which give a clear explanation of the functionality we expect of our agent.

User stories of features

As a housing corporation agent
I want to know if ground is going to be available soon
So that I can consider buying the ground

As a housing corporation agent
I want to buy ground
So I can build residential buildings

As a housing corporation agent
I want to build residential buildings on ground that I own
So that I can have buildings to sell

As a housing corporation agent
I want to sell residential buildings I built
So that I can make profit

As a housing corporation agent
I want to be able to renovate my buildings
So that I can modernize my neighborhood

As a housing corporation agent
I want to propose ideas to other parties
So that I can build a better neighborhood

As a housing corporation agent
I want to be able to judge incoming propositions
So that I can accept or decline them

As a housing corporation agent
I want to be able to use indicator metrics
So that I can evaluate my performance

As a housing corporation agent
I want to make intelligent propositions
So that I can improve my indicator metrics

User stories of know-how acquisition

As a developer
I want to know the functionality of the Tygron Engine
So that I can develop my agent to use the features of the engine

As a developer
I want to know how I can work with the Tygron Connector
So that my agent can be connected to the Tygron engine

As a developer
I want to know how to work with GOAL
So I can make a GOAL agent

As a developer
I want to know how the EIS Connector works
So I can successfully connect my agent to the Tygron Engine

4. Definition of Done (DoD)

In this chapter we will talk about when we consider a product done at the end of each sprint. Each sprint will be of a one week length. It is important to have a working product by the end of every sprint so that we can acquire feedback and maintain agility.

A feature is considered "done" if and only if all of the following points apply:

- All of the requirements for the feature are met.
- The feature has been thoroughly tested.
- All old features which at one point in time worked must still function correctly.
- The code has been documented:
 - The documentation has been made available to all of the members of the team (either on GitHub or Dropbox).
- The developer of the code has committed the code to the GitHub repository
- The code has been merged into the dev or master branch after the pull request has been approved.
- The customer is satisfied with the feature.
- When all of the above points are achieved, the feature will be marked as done during the scrum meeting.
- If a feature is, for any reason, aborted before completion, all documentation/code should be removed to ensure a clean working environment.
- If a feature has to be removed after it has been completed, the team will decide what to do as soon as possible after the decision has been made.