

Interaction Design

Michiel van den Berg 4391039 michielvandenb
Stefan Breetveld 4374657 sbreetveld
Timo van Leest 4423798 timovanleest
Daan van den Werf 4369556 djvanderwerf
Job Zoon 4393899 jzoon

May 30, 2016

Contents

1	Persona	3
2	Usage scenario	4
3	Product design	5
3.1	Analysis	5
4	Glossary	6

1 Persona

This section describes the software architecture views, it explains the sub-system decomposition, the hardware/software mapping, persistent data management and the concurrency.

Hank - Property development

Hank is someone who is in the planning phase of a construction project, for a property development company, somewhere in The Hague. He wants to use the Tygron engine for detailed information on possible outcomes of scenarios that can happen during his or her project, because the Tygron engine can provide him with this decision before he actually has to spend any money on the actual project.

He has his own login for the Tygron engine, which he can use to create projects, virtual versions of the area surrounding his project location. In these projects he can specify all kinds of rules or laws that governments or other parties have in effect in that area. He can also create Indicators. Indicators can help him visualize his current goals and the state of these goals, they also provide some sort of score system. This score system will be shown at the end of the planning session and shows you how much of your own goals you completed. One of his goals is to make at least €1.000.000 profit from his project in the following 5 years.

Lisa – Municipality

Lisa works for the municipality that houses hank's project. She wants to make sure that Hank can complete his project, but he can't break any laws with his project otherwise she has to say no. Since Hank and Lisa need to work together, Lisa also gets a login for the Tygron environment. She can use this login in the same way hank can, but instead of creating new projects she edits hank's projects. She add indicators of her own, like not going bankrupt and having a set amount of square meters of green per square meter of usable building space.

2 Usage scenario

The Tygron engine is designed to use in scenarios where all parties involved with the project for instance: the user (project lead), municipalities, housing cooperatives, etc. Each party makes decisions about actions that involve their party, for instance municipalities need to decide if you are allowed to construct a building, if the building is to tall, if you can buy a piece of land for your project, etc. But what if you decided that you wanted to try something in the Tygron engine, but not all other parties are available, you then have to either find someone who can fill in for the other parties or you postpone to a later date.

Our project can help with this by replacing a real world party with a computer controlled party. This computer controlled party, from now on referred to as agent, is designed to follow the guidelines that it is given through the engine, but it can also negotiate with other parties, be they human or computer controlled. The actions these agents take aren't necessarily 100 percent accurate to what the party would actually do, but we try to make it as close as possible.

3 Product design

3.1 Analysis

The main problem we want to work on is to make it possible for users to use the Tygron engine while not all other parties are present or ultimately when a user is alone. We want to achieve this by creating agents that will act like the party they replace. We will connect to the Tygron engine by using their SDK.

We are to use Goal as our platform for creating these agents. Goal is designed for creating (somewhat) Artificial intelligence. It has easy goal management and also has a clear system that acts like the memory of the agent. This memory, also called the believe base, can also store any information the agent receives.

One of the limitations of goal is how it interacts with other software. We were supplied with a “connector” for this that will handle the communication between goal and the Tygron SDK. Another limitation is its ability to handle mathematical problems. Therefore it can be difficult for the agent to figure out where to construct a new building. This Limitation can be overcome by adding a new action to the “connector” which specifies where it can build.

4 Glossary

Agent

An agent is an autonomous entity which observes through sensors and acts upon an environment using actuators and directs its activity towards achieving goals.

Engine

An engine is a type of software that generates source code or markup and produces elements that begin another process, allowing real-time maintenance of software requirements.

Connector

A software solution that enables the GOAL agents to connect to the SDK of the Tygron environment.