Product Vision

04/05/2016

CodeFox

Marco Houtman
Ronald van Driel
Joshua Slik
Matthijs Halvemaan
Lisette Veldkamp

Product vision

The product vision provides with the main guidelines which we strive to adhere to throughout the project. It represents the core of our perception on the product we will deliver during this project.

Target customer

We view the company Tygron as our client for this product. [1] Tygron is the provider of ‘Tygron Engine’ which is a serious game used for online 3D project planning for urban communities. This game is multiplayer and requires several players to plan, negotiate and make decisions within an urban community.

Customer needs

Our client request players to be replaced by computer driven virtual humans, so that they are able to run a lot of simulations and see the outcome every time. We use the MoSCoW requirements method to prioritize the features.[2]

Must haves

The virtual humans must interact within a predefined environment, which therefore has to be constructed as well. A virtual human must be able to successfully interact with human players and other virtual humans by manufacturing and proposing feasible plans and accepting or declining proposed requests.

Should haves

A virtual human should behave as closely to how a real human would behave in the environment.

Could haves

The data from the resulting final situations and the decisions made to come to that situation could be visualized in such a way to give more insight at a quick glance what decisions are made often, and thus what decisions could be the best.

Won’t haves

The virtual human won’t be able to interpret natural language and communicate with real humans in this manner. It also won’t be able to access online data to gain additional information.

Product competitors

Our product will be used to replace the existing need for real humans to play in Tygron Engine. It can be time consuming, expensive and unfeasible to get humans to simulate an urban community. With our product however these simulations can be remotely initiated and played out, which requires almost no human interaction.

We have analysed alternatives for our product in the context of replacing human interactions with artificial intelligence will be compared against our product.

For example we analysed the Monte-Carlo tree search method for decision making in artificial intelligence. This method uses a search space containing actions and bases actions

on past experiences and interactions using backpropagation. [3] This algorithm only starts using more refined moves after it has used multiple randomly chosen actions. Our product however will instantly be able to interact successfully and intelligently with its environment.

Another similar context in which multi-agent systems have been used is described in an international workshop about agent technology for disaster management. In a related article the following is mentioned about the agent system: “In joining RoboCup rescue community,we have recognized that rescue agents' behavior has been analyzed ad hoc and evaluated by employing various standards.” [4] This implies that the different agent types are not unlike our virtual humans continuously evaluated based on several performance metrics. We want to have our virtual humans permanently evaluate their decisions with respect to the performance metrics.

Our product is different from other implementations of agent systems because it continuously develops and evaluates urban planning decisions and is able to interact directly within a Tygron Engine environment.

Timeframe and budget

For the entire development of the product ten weeks are dedicated. The team has been provided with a framework to develop our virtual human which is, in this case, GOAL. [5] To connect the virtual humans to Tygron Engine a specific connector and an interface for the environment is provided as well. Besides the given tools there is no additional budget available.

References

1. Tygron NL, *Tygron Engine*, version 2016.5.0.0, Dutch, Tygron (Saturnusstraat 60: Tygron NL, 2016), http://www.tygron.com.
2. Haughey, Duncan. "MoSCoW Method." *Retrieved October* 28 (2000): 2012.
3. G.M.J-B Chaslot, “Monte-Carlo Tree Search” (Universiteit Maastricht, 2010).
4. Nicholas R. Jennings et al., “First International Workshop on Agent Technology for Disaster Management,” 2006, https://www.researchgate.net/profile/Toru_Ishida2/publication/238684507_First_International_Workshop_on_Agent_Technology_for_Disaster_Management_Foreword/links/0deec5319dd07ba9d3000000.pdf 7: p.78.
5. Hindriks Koen, *GOAL*, version 2.0.1.201604181354, Dutch, accessed April 5, 2016, http://ii.tudelft.nl/trac/goal.