Development of a real-time driving simulator with external artificial intelligence interaction with the purpose of allowing students to practice writing artificial intelligence

David Lambeth

BSc (hons) Computer Science

2018

This dissertation may be made available for consultation within the University Library and may be photocopied or lent to other libraries for the purposes of consultation.

**COPYRIGHT**

Attention is drawn to the fact that copyright of this dissertation rests with its author. The Intellectual Property Rights of the products produced as part of the project belong to the author unless otherwise specified below, in accordance with the University of Bath’s policy on intellectual property (see http://www.bath.ac.uk/ordinances/22.pdf). This copy of the dissertation has been supplied on condition that anyone who consults it is understood to recognise that its copyright rests with its author and that no quotation from the dissertation and no information derived from it may be published without the prior written consent of the author.

**Declaration**

This dissertation is submitted to the University of Bath in accordance with the requirements of the degree of Bachelor of Science in the Department of Computer Science. No portion of the work in this dissertation has been submitted in support of an application for any other degree or qualification of this or any other university or institution of learning. Except where specifically acknowledged, it is the work of the author.

………………………….

# Abstract

Contents

[1 Abstract 4](#_Toc508560998)

[3 List of Figures 6](#_Toc508560999)

[4 Acknowledgements 7](#_Toc508561000)

[5 Introduction 8](#_Toc508561001)

[5.1 Goals 8](#_Toc508561002)

[5.2 Motivation 8](#_Toc508561003)

[6 Literature and Technology Survey 9](#_Toc508561004)

[7 Requirements Analysis and Requirements Specification 10](#_Toc508561005)

[7.1 Establishing Requirements 10](#_Toc508561006)

[7.1.1 Questionnaire 10](#_Toc508561007)

[7.1.2 Existing Solutions 10](#_Toc508561008)

[7.1.3 Use Cases 11](#_Toc508561009)

[7.2 Requirements 13](#_Toc508561010)

[7.3 Functional Requirements 13](#_Toc508561011)

[7.4 Non-Functional Requirements 13](#_Toc508561012)

[8 Design 14](#_Toc508561013)

[9 Detailed Design and Implementation 15](#_Toc508561014)

[10 System Testing 16](#_Toc508561015)

[11 Conclusions 17](#_Toc508561016)

[12 Bibliography 18](#_Toc508561017)

[13 Appendices 19](#_Toc508561018)

# List of Figures

[Figure 1 - Use case showing a teacher creating a race 13](#_Toc508559915)

[Figure 2 - Use case showing a teacher viewing a race 13](#_Toc508559916)

[Figure 3 - Use case showing a student viewing a race 13](#_Toc508559917)

[Figure 4 - Use case showing an AI updating a car's controls 14](#_Toc508559918)

# Acknowledgements

# Introduction

Artificial Intelligence is an incredibly large field with far reaching applications in everyday life as well as niche situations. Because of this importance there is a massive need for incoming developers to be able to understand and utilise artificial intelligence within their roles.

## Goals

## Motivation

# Literature and Technology Survey

# Requirements Analysis and Requirements Specification

## Establishing Requirements

There was no stakeholder used for this project, instead requirements were gathered from various sources. Existing solutions and use cases were the largest influencers of requirements, a questionnaire was also used.

### Questionnaire

See Appendix #. The questionnaire was designed to make use of people who had previously been taught AI, hoping to get information on the existing solutions and what teachers are currently using. Teachers and students who have used other solutions are one of the best sources for requirements as they have already made opinions on the solutions that they have available.

The results of the questionnaire (Appendix #) didn’t give too many suggestions of what should and shouldn’t be included, though it did show that the educational sector appears to use tools that are only used by a single person at a time, on their own machine.

The initial results were hard to use as the first respondents had not been part of any AI courses and so did not have much to add to the results. The respondents that have been included in the results are the ones that had some useful insight to add.

### Existing Solutions

#### Top Down Racing Games

Some existing solutions were found through the questionnaire, these were then played to get a feel of their features. Other solutions were ones already known and these were analysed in the same way.

Key features or attributes that most or all these solutions had in common were:

* Countdown to start
* Music during setup
* Music during race
* Finish screen, showing the times of each car
* Position indicator, showing what place you are in
* Keyboard controls were roughly the same between games
* The number of cars in the race was not variable

Some features won’t be taken over, such as the number of cars not being variable, as this doesn’t fit with the motivation of having a classroom situation, where there could be less or more students.

Features such as a countdown will be in the requirements as these are necessary for the function of the game.

#### Web Browser Multiplayer Games

The games that were analysed for this project are LazerSharks.io and Littlewargame.

##### LazerSharks.io

LazerSharks.io is a simple game where you play as a shark with a laser attached to its head (LaserSharks.io | Play LaserSharks.io for free on Iogames.space!, 2018). The aim of the game is to eat the little fish that live around the game world whilst avoiding being killed by other sharks. The main thing taken from this analysis is how quickly the game is joined, there is no lengthy setup or signup process; you simply choose a nickname and a colour, and you can begin.

##### Littlewargame

Littlewargame is a strategy game where you build, upgrade and battle with others or with computer players (Littlewargame, 2018). Again, the entry to the game is minimal, though setting up a game for yourself rather than joining one takes slightly more setup which gives you more customisation. The features taken from this are the customisation, where you can choose what map you want to play on, and the option of having computers to play against.

#### Educational AI tools

##### LEGO Mindstorms NXT

The LEGO Mindstorms NXT kit is a buildable and programmable robot generally used for educational purposes (Home - LEGO.com, 2018). The kit includes a programmable brick that can be used to control the robot in its movements.

As the robot is programmable and comes with sensors to perceive the environment, it provides an excellent platform for teaching AI, giving a clear display of how AI acts in a real-world scenario.

There are already courses that use the LEGO Mindstorms NXT kit to assist in teaching AI or testing knowledge. One of these is at the University of Bath for final year students, assigning a single robot to a pair for a coursework (Programme & Unit Catalogues - University of Bath, 2018).

Another course is at FunTech, which is an extracurricular academy teaching computing, their course uses the robots to teach about path-finding (Lego NXT | Lego Robot Programming For Kids | Lego Holiday Camps, 2018). The course ends with a maze solving tournament between students, this lets them see the advantages and disadvantages of different approaches.

Both courses use the robots to give a visible indication of how the AI is performing. Requirements taken from this are the visual and real-world applications, also the need for a solution that allows many students to participate together at the end.

#### External AI tools

##### OpenAI Universe

OpenAI’s Universe project allows an AI to play any number of games that have been ported into the project. Each game is held within a Docker container that has a standard interface for the AI to get the output video from the game and to give the inputs chosen (Docker, 2018).

This solution gives the AI the ability to play many games using the same interface, making it possible to create an AI that is capable of playing lots of different games. The requirements gathered from this all relate to the externalised nature of the AI. Having AI that is part of a larger system, such as during developing a game and adding in AI for the enemies, means that you have to be creating a larger system. Having a system that is already created and allows for you to make a small AI that communicates with it, such as using a Docker container, removes the need to be making something larger.

### Use Cases

Using the questionnaire and existing solutions, use cases were created. The users are a teacher, many students and an external AI.

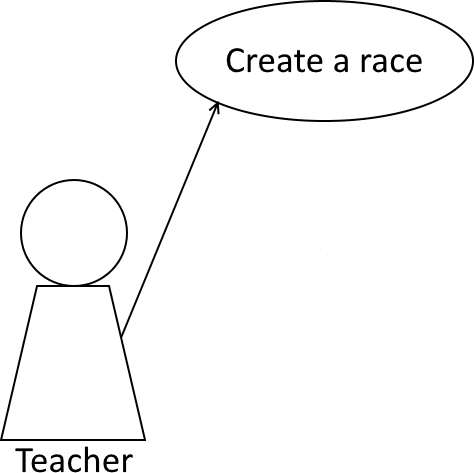


Figure 1 - Use case showing a teacher creating a race

It is assumed that the user in the teacher position will be creating the race, including all setup.

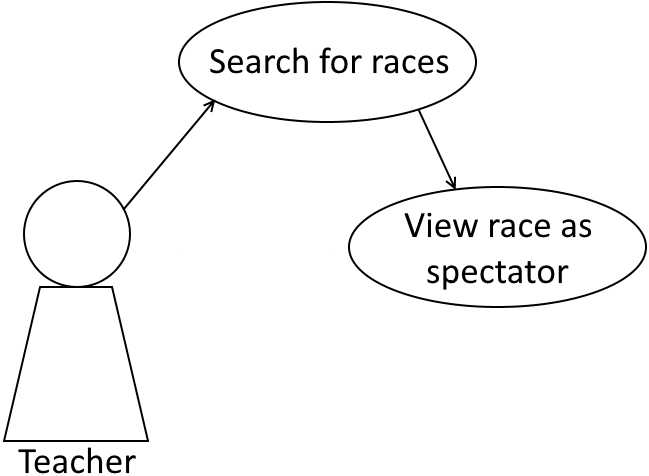


Figure 2 - Use case showing a teacher viewing a race

The teacher will then be able to view the race as a spectator, which will allow the race to be shown on the main screen for all to see.

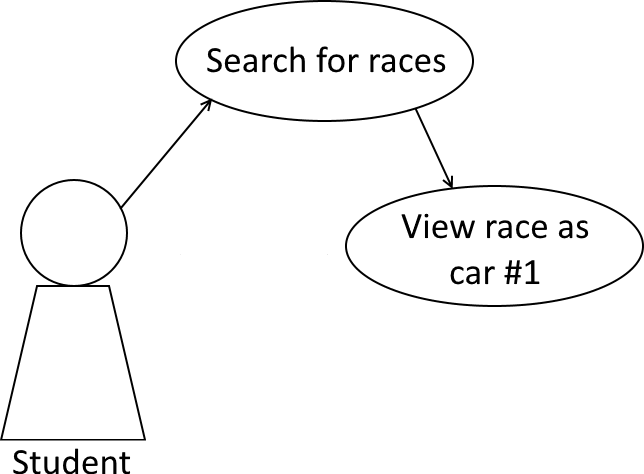


Figure 3 - Use case showing a student viewing a race

The student will be able to view the race from the perspective of a single car, with the camera following the car around the racetrack. At this point the student could drive the car using a keyboard, or just watch as the external AI races.

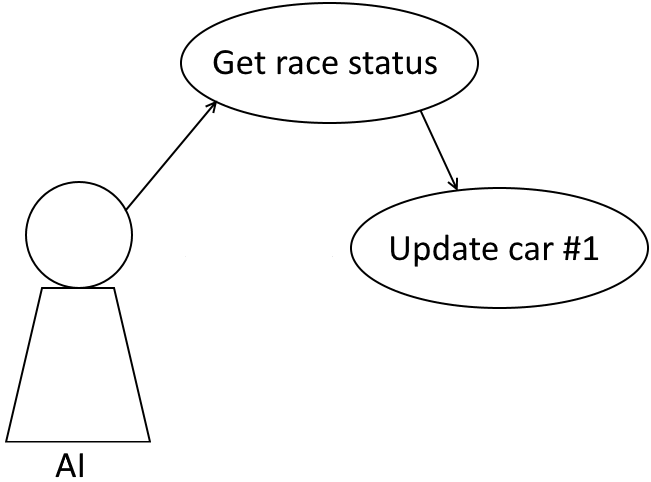


Figure 4 - Use case showing an AI updating a car's controls

The most important case, the external AI being able to update the car’s controls to drive it around. The AI must be able to get the status of the race, including all the information about the car’s location and where it is going.

## Requirements

Each requirement has been given a number that shows its section and exact identity, this will be used as the GitHub commit name so that it is easy to see what requirement is being worked on.

The requirements will be given a description and either must, should or could, to show the importance of the requirement on the system. Any requirement with must, has to be included and working for the system to be complete.

## Functional Requirements

Functional requirements describe the actual operation of the system, detailing what the system must, should and could do.

## Non-Functional Requirements

Non-functional requirements describe criteria for checking a system’s operation, instead of describing exactly what the system does.

# Design

# Detailed Design and Implementation

# System Testing

# Conclusions

# Bibliography

Anon, 2018. *LaserSharks.io | Play LaserSharks.io for free on Iogames.space!*. [online] Iogames.space. Available from: http://iogames.space/lasersharks-io [Accessed 11 Mar. 2018].

Anon, 2018. *Littlewargame*. [online] Littlewargame.com. Available from: http://www.littlewargame.com/ [Accessed 11 Mar. 2018].

Anon, 2018. *Home - LEGO.com*. [online] Lego.com. Available from: https://www.lego.com/en-gb/mindstorms [Accessed 11 Mar. 2018].

Anon, 2018. *Lego NXT | Lego Robot Programming For Kids | Lego Holiday Camps*. [online] Funtechsummercamps.com. Available from: https://funtechsummercamps.com/course-descriptions/lego-robotics-level-1 [Accessed 11 Mar. 2018].

Anon, 2018. *Programme & Unit Catalogues - University of Bath*. [online] Bath.ac.uk. Available from: http://www.bath.ac.uk/catalogues/2011-2012/cm/CM30229.htm [Accessed 11 Mar. 2018].

Anon, 2018. *Universe*. [online] OpenAI Blog. Available from: https://blog.openai.com/universe/ [Accessed 11 Mar. 2018].

Anon, 2018. *Docker*. [online] Docker. Available from: https://www.docker.com/ [Accessed 11 Mar. 2018].

# Appendices