horizontal line

Grand Traffic Auto

System Overview

**24th October 2019**

**Project Team**

|  |  |
| --- | --- |
| **Name** | **Student Number** |
| Robert Pennefather | 21511164 |
| Daniel Gonzalez | 22024722 |
| Martin Porebski | 21498791 |
| Jacqueline Soon | 21719676 |

Grand Traffic Auto is intended to be used in experimental economics based around traffic management. This project aims to build an initial framework that will allow researchers to replace classical, paper-based polls with immersive simulations that are believed to deliver more realistic results. The data collected from these simulations are particularly useful when researching and developing new solutions to traffic-based problems on our roads.

As a result, this group designed and developed a 3D game and a website. This framework is a starting point and therefore, it was constructed to be modular and flexible, which will allow future engineers to further develop it.

The game was built using Unity game engine (C#). It consists of a login screen, the configurable experiment (playthrough) and the end screen. Unity allowed for shorter development time while it also made it adaptable and modular. These qualities made it flexible and easier for future developers to pick up the development process from where we left off. Unity took care of low-level components such as graphical interfacing, assets management and OS integration. This allowed us to focus on the development of the game. The source code can also be compiled to run across multiple platforms (regardless of the OS). The final build was compiled using WebGL such that the game could be hosted online.

The website was built with ASP.NET (C#) framework and it was hosted using IIS on a Windows 10 VM located at UWA. This was done to speed up the development time as well as to keep all the products within the Microsoft suite to make it easier for future developers to learn and continue this project. The website consists of a home page, which can either contain a form with an initial survey or a link to an externally hosted one (e.g. on Survey Monkey). After completing the survey, the page redirects to the (WebGL) game hosted on the same server. The website also has a configuration page, which allows admins to easily configure the settings of the game for each experiment (playthrough). This is done by creating an encoded string from the admin’s input, which is then transferred into the game on start up.

Finally, MongoDB was used to collect and store all of the data across the framework. It was hosted on the cloud service Atlas. This allowed for always backed-up storage of the information that is easily accessible from anywhere. The data can be written or read through API calls or using the Atlas website, which provides an easy to use interface for the database.

**Source Code**

* Website:
  + GitHub:   
    <https://github.com/ok-martin/GrandTrafficAuto>
  + Access: David Glance, Chao Sun
* Unity Game
  + Unity Cloud: <https://developer.cloud.unity3d.com/orgs/uwa_grandtrafficauto/projects/grand-traffic-auto/>
  + Access: David Glance, Chao Sun

More information about accessing the source code is available in the Maintenance Manual.