Great Instructor Races

“You will RACE but will you ACE???”© (N)\*everLasting

Technical Document   
(Homework No.3)

Project team: (N)\*everLasting

Instructor: Dr. Araz Yusubov

Submitted in partial fulfillment of the requirements of the CSCI 4836: Game Development Fundamentals course project

|  |  |
| --- | --- |
| Version date | Version information |
| 18.11.2018 | Initial draft |
| 18.11.2018 | In this version, we have described the prototype that we have already done. |

|  |  |
| --- | --- |
| Other documents in the package | |
| File name | Brief description of the document |
| Audio.zip | Section 3, promised audio files. |
| Scripts.zip | The coded which we made our Class Diagram. |

|  |  |  |
| --- | --- | --- |
| Team member | Contribution to this homework (NOT the project) | Estimated % |
| Ilyas Karimov | Contribution to 2,5 | 25% |
| Sevil Jafarli | Contribution to 3,5 | 25% |
| Aytan Alakbarova | Contribution to 4 | 25% |
| Ismayil Bagirov | Contribution to 5. | 25% |

# Table of Contents

<Automatically generate here using Microsoft® Word menu References→Table of Contents>

# Introduction

This is part of the Game Design Document for a hypothetical project Great Instructor Races submitted for partial fulfillment of the requirements of the Game Development Fundamentals course in the School of Information Technologies and Engineering at ADA University, Baku, Azerbaijan.

**Bonus:** Get extra **5%** (x2) points for revising and SUBSTANTIALLY improving each of the Homework 1/2 content.

All Microsoft© Word documents should be submitted as separate files. Any additional files of other types e.g. diagram and charts will usually be inserted to this document as embedded images, but the source files e.g. Photoshop .psd files should also be submitted. In this case (of having multiple non-Word files) all of them should be submitted as part of a SINGLE .zip archive file.>

## System Requirements

While developing a game, for every game, developers write the game(codes) through the applications with any existent versions(such as Unity 2017.2.15f2, Xcode 9.0) by which the system requirements are automatically detected or manually chosen. The reason of this action is to achieve a better performance so that players will not experience any kinds of issues while playing. Since in newer version applications it can be made heavier games with novel features, the system requirements should be there to restrict players with old hardware and software. In our case, as developers, we did not restrict the users by our requirements, however, the applications which we have utilized have its own requirements and they are mentioned down below:

**For developments**

**OS**: Windows 7 SP1+, 8, 10, 64-bit versions only; macOS 10.11+

Server versions of Windows & OS X are not tested.

**CPU**: SSE2 instruction set support.

**GPU**: Graphics card with DX10 (shader model 4.0) capabilities.

The rest mostly depends on the complexity of your projects.

**Additional platform development requirements:**

* iOS: Mac computer running minimum macOS 10.12.6 and Xcode 9.0 or higher.
* Android: Android SDK and Java Development Kit (JDK); IL2CPP scripting backend requires Android NDK.
* Universal Windows Platform: Windows 10 (64-bit), Visual Studio 2015 with C++ Tools component or later and Windows 10 SDK

**For running Unity games**

Generally content developed with Unity can run pretty much everywhere. How well it runs is dependent on the complexity of your project. More detailed requirements:

* Desktop:
  + OS: Windows 7 SP1+, macOS 10.11+, Ubuntu 12.04+, SteamOS+
  + Graphics card with DX10 (shader model 4.0) capabilities.
  + CPU: SSE2 instruction set support.
* iOS player requires iOS 8.0 or higher.
* Android: OS 4.1 or later; ARMv7 CPU with NEON support or Atom CPU; OpenGL ES 2.0 or later.
* WebGL: Any recent desktop version of Firefox, Chrome, Edge or Safari.
* Universal Windows Platform: Windows 10 and a graphics card with DX10 (shader model 4.0) capabilities

## Concerns and Alternatives

There’s always a risk in every operation(surgery) in medicine, but the probability of the risk depends on the hardness of the surgery, and the same goes for the development of the games. we can categorize the hardness of our game as average and mention that we experienced some hardships while developing it.

Technical Issues:

The first problem was to develop/find the relevant models since Unity Community Store is so limited with several of them and the models which were planned to be in our game do not exist or are not free of charge(they are paid) and we are reluctant to pay. The second problem was that some of our developers have different versions of Unity, which also caused some internal problems(graphical problems) while running the game. Some of the assigned colors were demonstrated differently, the color white was mixed with black, green and white and we did not find the cause of it. The third problem was related to photoshop, we rate its performance as mediocre and mention that we experienced some freezes/crashes while using it in the process of image editing and transforming.

Alternatives:

The first concern was time management since we are senior students and have limited time with different schedules, consequently, meeting, discussing and agreeing upon one unique conclusion took oodles of time. The second was related to assigning the written codes to one unique workplace since we have many individuals working on the code. Writing them individually and merging it was also extra time. We may also belong this issue to our document-writing because in the beginning, one person(the leader) should assign the tasks to group members and in the end, took all different parts and merge them and ensure the consistency exists.The third was the cartoon-based pictures of our instructors. In order to achieve high quality, we should find the high quality pictures of our instructors so we can transform it to cartoons, nevertheless, with the help of internet we are still incapable of it and through the photoshop, we were unable to achieve that quality.

## Resources

* **Unity Personal 2018.2.16** (game engine)
* **GitHub Inc** (hosting service for version control)
* Photoshop CC 2016
* **C#** programing language (Visual Studio IDE)
* Youtube Unity tutorials
* Unity Store

**A close up of a map

Description automatically generated**

# References

<Insert here any document referred to in the document. An example might be articles or Web sites that you consulted during the literature search. This is not just a list of used materials, so do not forget to clearly MARK the exact points(s) of reference in the main text.>

<https://unity3d.com/unity/system-requirements>