HATtrick Games

Team Charter

# Purpose

To make a polished, stable and fun 3D game over the length of a semester to improve our programming skills, work in a team environment and to add to our group portfolio. We would also like to develop a deeper understanding of game development techniques and processes used in the industry such as game design, storytelling and level design.

# Team Rules

* Do your job.
* Don’t commit to the main build without first checking that the program runs without errors.
* If you say you’re going to have something done by a certain time, make sure that you have it done by that time.
* If you are having trouble with something, ask the other team members for help. Sometimes a different point of view could solve the problem in next to no time.
* If you do not write the code, don’t change the code. Instead make a note of what you think should be changed and bring it up with the team member that wrote the code.

# Team Members

* Hamish Carrier
* Arran Ford
* Timothy Veletta

# Team Operations

## Accountability

* Hamish - Not available Tuesday
* Arran - Not available Tuesday
* Tim - Not available Tuesday

## Communication Methods

* Steam Chat
* Skype
* Google Docs for Task Management and other Documents

## Decision Making Process

The division of tasks will use the following steps:

1. Volunteer Basis
2. Group Discussion
3. Randomised distribution of less favourable tasks

## Conflict Resolution and Problem Solving

If a conflict cannot be resolved first by group discussion then the remaining member not in the conflict has the final say. If all 3 members are in the conflict then it will be resolved in a random civilization 1 v 1 v 1 fight on Age of Empires 2 HD.

# Resources

## Software

* Windows 7/8
* Visual Studio 2010
* 3DS Max 2012

## Libraries/Engines

* Ogre Graphics Engine
* Bullet Physics Engine

# Coding Standards

## Naming Conventions

* Class names should be nouns in UpperCamelCase, with the first letter of every word capitalised.
* Method names should be verbs in lowerCamelCase, with the first letter lowercase and the first letter of subsequent words in uppercase.
* Variable names should be written in lowerCamelCase and should not start with an underscore or dollar sign.
* Constants should be written in uppercase characters separated by underscores.

## Layout Conventions

* Write only one statement per line.
* Write only one declaration per line.
* Try to keep lines less than 80 characters long unless it impacts code readability.
* If a line is too long and is to be completed on further lines, indent those further lines one tab stop.
* Use parentheses to make clauses in an expression apparent. i.e.

if (**(**val1 > val2**)** && **(**val1 > val3**)**)

{

// then..

}

## Commenting Conventions

* Place comments on their own line, not at the end of a line of code.
* Begin the comment text with an uppercase character.
* Insert one space between the comment delimiter (//) and the comment text.

# Agreement

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Team Member Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Team Member Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Team Member Date