Toolsets

The toolsets that our group shall be using will be focused on the best tools to create a game and ways to facilitate easy sharing and collaboration. We considered many game engines, programming languages and sharing applications. After looking at the available options the choices we made were to use Unity as the game engine and maker, C# as the programming language and Git/GitHub to share and collaborate.

# Engines

Game maker

For

1. Easy to use
2. Prior knowledge of use by all
3. Simple sprite based graphics

Against

1. 3D difficult to implement
2. Limited programming language
3. Restricted free version

XNA Game Studio

For

1. C# is supported
2. Free commercial license
3. Integrated with Visual Studio

Against

1. Unfamiliar
2. Heavily code based environment
3. Only runs on PC and Xbox.

**Unity**

**For**

1. **Easy to use**
2. **Free commercial license**
3. **Graphic based workflow**

**Against**

1. **Large projects to sync of CVS**

UDK

For

1. Popular, well documented
2. High graphic quality
3. Good free license

Against

1. Uses custom programming language
2. Confusing interface/slow workflow

Cryengine 3

For

1. Very good graphical quality
2. Good interface
3. Good free license

Against

1. Not hugely popular
2. Suited mainly for 3D games

# Programming Language

JavaScript

For

1. Some experience
2. Simple to learn
3. Used in many other applications

Against

1. Not much experience
2. Type used in Unity is slightly different
3. Must use OOP

C#

For

1. Many other uses
2. Known to all group members
3. More documented than JS

Against

1. Not used in majority of games
2. Less efficient than C++
3. Not compatible on non-windows platforms.

C++

For

1. Known by all group members
2. Commonly used in the industry

Against

1. Not used in Unity.
2. Hard to port to other platforms

# Group Tools

Github

For

1. Has very good versioning
2. And branching
3. Works on non-mobile platforms

Against

1. Complicated
2. Not on mobile platforms

Dropbox

For

1. Free for up to 10gb per person
2. Available on all platforms
3. Easy to use

Against

1. Has no versioning
2. Or branching
3. Limit on storage