UNITY

Game engines:

- made from software framework and development tools

- handles common tasks

- Handles low- levels tasks

- handles platform differences – make the game and it will automatically compile into other devices e.g. from xbox to ipad

- Pretty much does everything for you

Examples:

* **Unity 3D**
* UDK
* Unreal

A lot of similarities between unity and other engines involved

If you want to use anything in unity it MUST be in your game folder. E.g. various images or characters.

Make sure you AREN’T in ‘play’ mode when editing- it will not be saved

Edit – preferences – colour – playmode tint colour (edits to interface colour when you are in play mode, stopping wasting time on editing just that play session)

Useful Links:

* unity3d.com/support/resources/downloads
* unity3d.com/support/documentation
* [wiki.unity3d.com](http://wiki.unity3d.com/index.php?title=Main_Page)
* forum.unity3d.com
* [www.google.com](http://www.google.com/) – *use this to search instead of the unity support page (documentation)*
* gamasutra.com – *games journalism website*

Basic concepts:

* Project – contains everything you need, will be a folder and everything inside will be your project, unity will already have pre-processed stuff for you – you cannot save a scene outside of the project folder. Copy the whole project folder to move it to another computer, when moving files within you project, only do so using unity as you could break the links if you do it using the windows search.
* Scene – A ‘Level’ of the game.

They are also used for: common GUI, menus, initialisation / Splash screens.

Building Blocks:

* Game object- entity or concept e.g. enemy or store of data which has no other users.

Always has a ‘transform’: position, size ect……

A container for components

Always has a hierarchy, so will be made easy to find everything you need.

Always has a name, transform, layers and tags ( all enemies can have one tag and then you can code they’re behaviours / looks)

* Component- Behaviours – a wall have many different components, its rendering, its physics, and its reaction to being shot/ hit.

-often have parameters e.g. health bar

* Asset – the content for your game modules, textures, sounds, materials.