CROSS-PLATFORM DEVELOPMENT PROJECT RESEARCH DOCUMENT

ASSESSMENT DESCRIPTION

For this assessment you are required to create a game using either Unity 3D or Unreal Engine 4, either individually or as part of a team, and build it for multiple platforms. Which platforms you support will depend on your available devices, but your teacher will advise you as to which platforms would be best supported. At a minimum, your game must perform on:

- At least two different web browsers, and
- At least two different digital devices

The game does not have to be a detailed game project but must at least make use of loaded assets and a basic Graphical User Interface (GUI) demonstrating industry best practice. You must also demonstrate the use of your target platform's specific input devices. For example, on a mobile device you would need to demonstrate *touch-screen input* whereas on a PC the project would utilize *controller input or keyboard.*

The project involves several stages of development, outlined below.

Briefly describe the cross-platform application, game or simulation you are researching.

(*This is your initial idea to focus your research. The application described in your design documents or your final build may end up being different from this description*)

The game I will be researching is a 3D Endless Runner. It will feature basic controls, procedural environment, player skins, scoring, and powerups. Moreover, there will be a GUI for the main menu, options menu, pause menu, and player character select. There is a chance of an achievements menu. The visual style will be low poly to keep things smooth and simple, with the added benefit of increased performance.

List the software you will use to create your project.

(*Include any third-party plug-ins, APIs or libraries, if known.*)

The game will be built using Unity Engine. In order to test the project with various platforms, I will need some different software applications.
To test for mobile and handheld devices, I will use my personal Samsung a50 in developer mode coupled with a Samsung USB Driver and the Java Development Kit.
In order to deploy to both Google Chrome and FireFox. I will use Unity's inbuilt WebGL for testing, and in order to deploy I will upload my game to Unity Play and provide a shareable link to each browser.
As for a PC build, I have installed inno setup to release an executable of my game.
All assets will be sourced externally.
Doxygen will be used to generate documentation for the entire project.

With reference to the above list, what legislative frameworks or organisational standards govern the use of this software?

(*For example, include any End User Licence Agreements (EULAs), terms of service, copyright notices, licencing information, developer guidelines, coding standards, or similar.* *Include URL links where relevant.)

Unity provides a standard EULA. I have a Personal license for Unity which is eligible for any game i release, provided it generates "*Revenue or funding less than $100K in the last 12 months*"

Unity's asset store provides a Standard Unity Asset store EULA for every asset.
I don't believe the other applications I am using have any EULAs, at least based upon my research.

List the cross-platform installers and installation methods you will use, or the specific binary formats that are required to deploy the game.

(*This list should include all platforms you plan to deploy your game or application to. Your game or application must be deployed to at least two different web browsers, and at least two different digital devices – one of which may be PC*)

The game will be built for PC, mobile, and web.
I will deploy it as a PC executable, using the inno setup.
For mobile, I will be testing on a personal android device, using a samsung usb driver and the java development kit. I will release an android.apk build of my game.
For Web, I will use unity's inbuilt webgl to test the game in the Google Chrome and Mozilla Firefox web browsers, before releasing it to Unity Play as a public shareable link.

What IDE will you use?

(*Identify your reasons behind this choice.*)

I will be using both Visual Studio and Visual Code for editing and viewing scripts. I prefer to use VS for most things such as editing scripts due to its versatility. VC is preferred for viewing scripts without editing them.
Visual Studio comes with XML Comments built in, to which I will make full use of, however I will release all documentation using DOXYGEN, also adhering to the XML SUmmary Comments to ensure doxygen is used effectively.

Identify the cross-platform libraries, plug-ins, or APIs you will use.

(*Mention any restrictions or limitations that exist with these libraries on each target platform. For example, some parts of the .NET class libraries implicitly depend on threads, but some platforms (WebGL) do not support threads.*)

Bluestacks will be used in unison with a personal android device, in order to test the Android Platform Development.
Unity Play will be used to finally upload the WEBGL build, however testing will be done locally between unity and google chrome.

What issues exist, or do you expect might exist when developing for the target platforms you have identified?

I have several concerns such as incompatibilities between the different platforms. I'm concerned issues may arise, in particular, when deploying to android, that because I am testing on a rather unpopular device, it won't appear the same for the final product.

The differing platforms each have different capabilities and requirements which will need to be adjusted in order to accommodate to. One of the biggest concerns is GUI, ensuring that it scales and displays correctly regardless of screen size. Some handheld devices have a small border covering the edge of the screen (*or potentially cases*), ao allowing some buffer space will need to be considered in order to make GUI functional.

Further input methods will need to be implemented to accommodate handheld touch inputs, every button should register a touch input AND mouse click (*including pausing*). While PC and web builds can play using mouse and keyboard, while mobile needs to play by registering screen inputs.

I researched games such as Race the Sun, Alto's Adventure, Alto's Odyssey, and Temple Run, taking a lot of inspiration from them to guide the basis of my game. Race the Sun is available on console, pc, and mobile, while all the others were on mobile, so comparing how different games handle different styles will assist in the development process.

List any areas in your game where pre-written scripting packages could aid in development.
(*For at least one of these items, identify a package from the Unity Asset Store (or another source) that may be suitable.*)

I intend to use visual assets from the Unity Asset Store to create an environment for the player experience.
Low Poly Bird: Ultimate Pack - The main character will be a bird flying through landscapes
Low Poly Flowers - Cute flowers to be used as environment assets or collectibles
Low Poly Environment - Used to create the playable world, including obstacles

---

List the pre-written scripting packages or plug-ins you will use during development.
(*Include a URL for each package or plugin*)

Cinemachine - Will be used for cinematic transition through the projects menus and gameplay to enable smoother transitions.

List the game engine and any additional development tools you will use.

The game engine used will be Unity Engine.
Git and Github will be used to manage the version control and backup of the project.
