

Cross Platform Development – Project Research Workbook

This workbook will help you focus your research for your project.

Once you have answered these questions, use this information in your Technical Design Document.

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)

Simplistic maze game with two moveable characters, when one character is moved the other will move in the opposite direction. With the goal to have both characters escape the maze.

List the software you will use to create your project.

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

Unity 3D
Visual Studio 2022
Standard Assets library
Text Mesh Pro
LeanTween

With reference to the above list, what legislative frameworks or organisational standards govern the use of this software (including any third-party plug-ins, APIs or libraries).

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

(Information in the AIE Student Handbook may also be relevant in relation to the use of software on campus machines.)

Include URL links where relevant.

Unity Terms of Service, <https://unity3d.com/legal/terms-of-service>

Visual Studio Terms of Service, <https://visualstudio.microsoft.com/license-terms/>

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)

PC: The game can be marketed, distributed and installed using a variety of programs that specialize in this field. Options include steam and the epic launcher.

Web: Installers on the web include google chrome, safari, microsoft edge and any other browser.

Mobile:

Games that are ported to mobile have limited options in terms of installers. These usually consist of using the google play store and the apple store.

What IDE will you use?

Identify your reasons behind this choice (ignoring the pre-configured environment on the campus computers).

For the purposes of creating our project, our team will use Visual Studio 2022 as everyone in the team already has experience using this software. Visual Studio also has many extensions and tools designed for Unity that make creating a game easier.

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.

PC Restrictions:

Games on this platform require an external controller.

Mobile Restrictions: No external controllers (e.g. keyboard, controller). Hardware limitations. Screen size.

Web Restrictions: WebGL doesn't support multi-threading and will block the main thread when decompressing the asset bundle. Cursor locking is restricted on safari and doesn't allow keyboard input on full-screen applications. Chrome requires large amounts of memory

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

When developing for PC, without using the mouse for controlling the characters the movement will be stuck to 8 directions at most (left, right, up, down and diagonals).

When developing for mobile, the method for moving the characters needs to be reworked as by default there is no way to interact with the game aside from touch screen. Scaling the content and anchoring will also need to be taken into consideration as it is a much smaller screen compared to other platforms.

When developing for WebGL, there are many restrictions to take into account

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.

Camera: Setting up smooth camera movements can be tedious and require a lot of tweaking. Having a pre-written script can save hours of work. Cinemachine is a great example of a prewritten camera controller.

Character Controller: Character controllers can take a while to set up and usually consist of the same settings and inputs. Having a pre-written character controller from an already existing project saves a lot of time.

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

Cinemachine - <https://unity.com/unity/features/editor/art-and-design/cinemachine>

LeanTween - <https://assetstore.unity.com/packages/tools/animation/leantween-3595>

Standard Assets Pack - <https://assetstore.unity.com/packages/essentials/asset-packs/standard-assets-for-unity-2018-4-32351>

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

Unity Game Engine

Visual Studio