**TEMASEK POLYTECHNIC**

**SCHOOL OF INFORMATICS & IT**

**DIPLOMA IN GAME DESIGN AND DEVELOPMENT**

**ASSIGNMENT BRIEF FOR PROGRAMMING WITH GAME ENGINES (CGE2C19)**

# Assignment 2: Practical Assignment - Presentation (20%)

For this assignment, you will:

## Make a Video Presentation

**Submission Date: 23 November 2023 - 2100 hrs**

You are to research and find the common causes of performance issues in computer game programming related to your presentation topic. You will then make a video of your presentation and submit it via a YouTube link.

The presentation topics are in Annex A in no particular order. You will receive your topic for presentation on Week 3 through a random selection process.

Your presentation must include the following:

* Definition of the problem statement – explain what the problem is that you are trying to explore.
* The root cause(s) analysis of the problem – explain the possible causes of the problem.
* The tools for root cause(s) analysis – list the tools you might use to analyse the root causes of the problem.
* Approach to resolve the problem based on the root cause(s) analysis – provide a set of solutions for solving the problem.
* Conclusion – conclude your findings.
* References – provide references.

To prepare for the presentation, you will have to do self-directed learning. You will have to seek and consult reference learning materials on your own. There are many online resources for performance optimisation in Unity. A good starting point is to look at the official Unity sources, such as the API documentation, Unity manual, Unity GDC, and other conference videos. You can also consult books from the library. Provide a list of all references used in your presentation.

**Annex A: Topics for Presentation**

|  |  |  |
| --- | --- | --- |
|  | **Title** | **Topic** |
| **1** | **File Fury** | Loading assets from the file system is a bottleneck. If not done correctly, time and resources are wasted. Do not get furious with your file handling! What are some file-handling best practices for Unity? In addition, Unity has a Resources folder. What problems can arise with this? What are some best practices for using the Resources folder? |
| **2** | **Loop De-loop** | You should know about Big O notation. Loops within loops can slow your game down. Unity has coroutines. Can coroutines sometimes help your loop performance? Are for and foreach loops equivalent? Are all data structures the same, performance-wise? What other guidelines can you give about iteration in Unity? |
| **3** | **Media Frenzy** | Games use many media assets. What aspects of your media must you be aware of? What optimisations can you make to your various media? How can you ensure your media does not massacre your performance? What tools are available to help you make the necessary changes to your media files? Note: This topic is about media asset files before they are loaded into Unity. |
| **4** | **Drawcall destruction** | Games are visual. Everything visible in the game must be rendered to the screen. How is rendering done by Unity? What is a drawcall? How does this impact your performance? How can you optimise the drawcalls made by your game? What tools can help you identify the impact of drawcalls on your game? |
| **5** | **Memory Madness** | Your game has a limited amount of memory to run in. How can you optimise the memory demands of your media? What profiling tools are available for you to better understand how memory is used in your game? Note: This is different from (3) Media Frenzy. This topic is about memory use as your game runs, not the media asset files on disk. |
| **6** | **Garbage Ghosties** | When your game runs, objects are dereferenced or destroyed. These objects then become the prey of the garbage collector. What is garbage collection? How can garbage collection impact your game’s performance? What is a garbage collection spike? How can you improve the garbage collection in your game? What tools can help you identify garbage collection problems? |
| **7** | **Import Impact** | Your media assets, as files on disk, must be imported into Unity. Unity has many different import options for the different types of media. What are these options and why are they important? Also, when exactly are your media files imported? Are media files loaded when the game starts, when a scene changes, or when a particular media asset is needed? Or is something else going on? What are some best practices for importing media into Unity? |
| **8** | **Object Oblivion** | Your game is just a bunch of objects that exist in memory. Objects come and go. Objects take up memory. Creating and destroying objects takes up time. What about collections of objects? Can using structs instead of objects sometimes help? What are some best practices for object management in Unity? |
| **9** | **Component Crash** | Unity is component based. A gameobject in Unity is basically a container for different behaviours and properties implemented as components. You often need to access and use these components. What problems can arise? What are some best practices for using and accessing components in Unity? |
| **10** | **Profile Propaganda** | Experience is necessary to write optimised code. However, you also need hard data to see which parts of your code give the most problems. The process of analysing code to understand sections of code that creates performance issues is called profiling. What profiling tools are available for you to better understand the various performance bottlenecks in your game? |
| **11** | **GUI Gluey** | The Unity GUI has many problems associated with it. Look online and read all the complaints! How is the Unity GUI architected, and what are some of the main issues you must be aware of? What are some best practices for GUI implementation in Unity? |
| **12** | **API Attack** | Unity has a very extensive and powerful API. But sometimes, this can lead to lazy programming and lower performance. What are some of the problems and pitfalls of using the Unity API? What specific API calls might cause lower performance? What are some of the best practices for using the Unity API? |

## Penalty on Late Submission, Incomplete Submission & Plagiarism

All late submissions will be penalised following the three levels of lateness as stated below. Take note that plagiarism is a severe academic offence ([See TP Plagiarism Policy](http://www.tp.edu.sg/staticfiles/TP/files/studentportal/Plagiarism%20Policy.pdf)). All submitted works should be genuine and originated from you.

TP Plagiarism Policy Source:

<http://www.tp.edu.sg/staticfiles/TP/files/studentportal/Plagiarism%20Policy.pdf>

|  |  |
| --- | --- |
| Late < 1 day | 10% deduction from absolute mark. |
| Late >= 1 and < 2 days | 20% deduction from absolute mark. |
| Late >= 2 days | No marks will be awarded. |

#End of Document