

<Name-of-Software-Application>

**CS 230 Project Software Design Template**

Version 1.0

**Table of Contents**

**Document Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Comments |
| 1.0 | <06/16/2023> | <Sean Jones> | <Revised the Recommendations section> |

**Instructions**

Fill in all bracketed information on page one (the cover page), in the Document Revision History table, and below each header. Under each header, remove the bracketed prompt and write your own paragraph response covering the indicated information.

**Executive Summary**

The Game Room is wanting a web based game to be developed for their android only game, Draw It or Lose It. They are wanting this game to be able to run on multiple platforms. The idea of this game is to have multiple teams that they to guess a random picture drawn from a library. The game consists of four rounds at a minute each. If the current team cannot guess the picture, the remaining teams get 15 seconds to guess the image.

**Requirements ....**

*<* Please note: While this section is not being assessed, it will support your outline of the design constraints below. *In your summary, identify each of the client’s business and technical requirements in a clear and concise manner.>*

**Design Constraints**

* The game need mulitple teams with multiple people in each team
* The teams and Game need to have unique names so the game can check if the name is free or not
* There can only be one instance of this game running at a time
* The game needs to run on multiple platforms

**System Architecture View ....**

Please note: There is nothing required here for these projects, but this section serves as a reminder that describing the system and subsystem architecture present in the application, including physical components or tiers, may be required for other projects. A logical topology of the communication and storage aspects is also necessary to understand the overall architecture and should be provided.

**Domain Model**

One of the OOP principles that are largly in play with this diagram is inharatance. Game, Team, and Player classes gain what is used in the Entity class without having to rewrite code a lot. One other big thing we can see from this diagram is aggregation. Many classes have a referance to another class through a "has a" type of relationship, or as stated an aggregation relationship. The GameService "has a" Game, Game "has a" Team, and Team "has a" Player.



**Evaluation**

Using your experience to evaluate the characteristics, advantages, and weaknesses of each operating platform (Linux, Mac, and Windows) as well as mobile devices, consider the requirements outlined below and articulate your findings for each. As you complete the table, keep in mind your client’s requirements and look at the situation holistically, as it all has to work together.

In each cell, remove the bracketed prompt and write your own paragraph response covering the indicated information.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| **Server Side** | Mac has flexible terminal commands for configuring a server. They are also upgradable. This is one reason as to why they are popular in web hosting.  Characteristics:  web hosting and flexible terminal commands  Advantages:  Has various options for web browsing requirments as well as good upgradable equipment  Disadvantages:  Not the most preferred for web browsing | Linux has really good security as well as a low coster cost. Since this is mostly open sourse, security flaws are caught really fast. It is harder to setup due difficulty finding apps for this platform.  Characteristics:  One of the most preferred and more secure  Advantages:  Most flaws with this are caught before it can become an issue  Disadvantages:  Difficult to find and use applications | This one has the most software as compared to the others. It is high in resourses for creaing a server but due to it being used by a lot of people, this system is more prone to being attacked.  Characteristics:  This is more dominant to other platforms  Advantages:  Has a higher comfort level for user as well as lower loading  Disadvantages:  Tech support is not the best and it is easily susceptible to viruses | This would be the higest user count as to any of the platforms. This would also be cost effective but it is best to keep a server in a single place. It also has poor security.  Characteristics:  Portable and easy to use  Advantages:  Really good compatibility, easy to use, and low cost  Disadvantages:  Poor security do to it being a somewhat popular option |
| **Client Side** | The cost to use mac would be simular to what one would find on Windows. The issue is however that this would require more time and experience than what is required for Windows. Its still easier than Linux however. | To get the full use out of this would require a lot of time and experience. There is a plus side to its low cost however. | This is one of the easiest to learn as well as having minimum time requirments. The cost would be simular to Mac however. | This does provide with great flexibility for devs and clients alike. However, this would be more difficult to use than other platforms. |
| **Development Tools** | The main lauguage used on Mac would be Swift. Mac can run other languages but Swift would be the popular option. We could also use Java, Python, or Ruby. | Linux can work with most languages and IDEs.  VisualStudio, Eclipse, and Pycharm all work as well as there prefered languages. | Like Linux and Mac, this platform can run most langages and IDEs. You could even use Swift on this platform if you wanted to. VisualStudio might be the best however when it comes to support on this platform. | Many applications can be created on andriod and Swift on this platform. One thing you could use to develop would be the countless IDEs that are on this platform. This however isnt the easiest way to develop. Java, Python, and C++ all have IDEs and support on this platform. |

**Recommendations**

Analyze the characteristics of and techniques specific to various systems architectures and make a recommendation to The Gaming Room. Specifically, address the following:

* **Operating Platform**: Being that it has a lot of support with various software and the ablity to be learned easily, I bevieve Windows would be the prefect platform to use. The price t o preformance that Windows offers as well puts it above most of the compitition.
* **Operating Systems Architectures**: Windows allows for the use and access of the full system and specs while developing. You are able to access the GUI while accessing system resources at anytime.
* **Storage Management**: One thing that comes with Windows is a feature called Storage Sence that allows you to manage all files on your system as well as how much space they take up. You can add new folders for storage as well as have access to cloud storage at any moment. You get to choose where you want your files to go inside of the system as well.
* **Memory Management**: <Explain how the recommended operating platform uses memory management techniques for the Draw It or Lose It software.>

Windows has its own virtual address space for each 32-bit process, allowing up to 4 gigabytes of memory to be viewed. Each process has a 8-terabyte address space on 64-bit Windows. This allows for a lot of memory to be avalible for use in the game as to keep the software from having memory leaks or issues while being played. This allows for easy storage as well for information.

* **Distributed Systems and Networks**: One thing you would need to do to make sure you are able to get this to run on mutiple platforms would be to find software that works on all major platforms you want this game to play on. One language that works for all of these would be C++ so that would be the best language to write this in. You would also want to make sure that the servers would be able to effectivly communicate between platforms. Finding software able to do that is important. Microsoft Azure seems to be a good bet for this.
* **Security**: My recommendation for security would have to be Windows. One really big reason would be the build in anti virus that the OS has build in to it. While you could download one to any of these platforms, having a free to use build in anti virus in a nice bonus to consider.