

<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 | <mm/dd/yy> | <Your-Name> | <Brief description of changes in this revision> |

**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 Gaming Room is planning to create a game called "Draw It or Lose It," which is currently exclusive to Android devices. However, they now aim to make it accessible on various platforms, including web-based platforms. The objective of the game is to have multiple teams, each comprising several members, compete in four rounds, with each round lasting for a minute. In each round, a random image is selected from a library of images, and one team has to guess the image within the given time limit. If they fail to guess it, the opposing team members then have a chance to answer within 15 seconds.

**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**

In order to meet the requirements for this software, we need to form multiple teams with unique names, each consisting of several individuals. Our objective is to develop a game that runs on multiple platforms, including Android, Windows, Linux, and Apple devices. To achieve this, we may need to rewrite the code in Swift for Apple devices or explore ways to adapt the existing codebase to other platforms by leveraging different programming languages. Our aim is to create a single instance of the game that users can access across different devices, ensuring a seamless and enjoyable gaming experience. Through strategic application development, we can fulfill these requirements and enable users to easily check the availability of game and team names while enjoying the software on their preferred devices.

**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**

In this scenario, the Entity class acts as a superclass, providing common attributes like "name" and "id" to its subclasses: Game, Team, and Player. Through inheritance, the subclasses inherit or obtain information from the superclass. Additionally, the relationships between Team and Player, Game and Team, as well as GameService and Games are depicted as "has a" relationships, indicating that one class has a reference to instances of another class. This aggregation relationship allows us to understand that the GameService manages multiple games, the Games class is associated with multiple teams, and a team consists of multiple players. By utilizing UML, we can effectively visualize and represent these relationships between the classes and their instances.



**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** | Terminal commands are versatile tools that allow users to configure servers, access them, and make changes as required. These commands are widely used in web hosting due to their popularity and effectiveness. They possess specific characteristics that make them well-liked in the field of web hosting. | Mac systems offer similar features and are more cost-friendly compared to other alternatives. They are highly regarded for their security measures and are the preferred choice for web hosting services. The main advantage is that security flaws are identified and resolved before they can cause any problems, making it a reliable option. However, a disadvantage is that it can be challenging to find applications that fully support the specific requirements of web hosting.> | Mac systems have a wider range of software available compared to other operating systems. They dominate over other platforms and are considered a closed platform. The advantages include meeting high resource requirements, having shorter loading times, and providing a high level of comfort for users. However, Mac systems are more susceptible to viruses and may have poor technical support, which are the main disadvantages. | If the server remains stationary and can be located in one place, it is preferable. However, other devices offer better specifications. The more popular option is to have a server with high portability. The advantages include reaching a wider audience, improved compatibility, and cost-effectiveness. However, there are disadvantages to consider, such as limited compatibility with different smart mobile devices and lower security measures. |
| **Client Side** | Developing a compatible application for all web browser platforms and mobile devices requires expertise, time, and testing. It is essential to adapt the design and functionality to ensure a seamless user experience. Keeping up with technology standards is also crucial for ongoing compatibility and performance. | To ensure an application is compatible with all web browser platforms and mobile devices, maximum expertise and time are needed. While the cost should be kept at a minimum. The application development process requires thorough testing and adjustments. It involves adapting the design and functionality for a seamless user experience. Staying updated with technology standards is important for ongoing compatibility and optimal performance. | To ensure an application is compatible with all web browser platforms and mobile devices, the application development process requires a minimum amount of expertise and time. The cost is comparable to that of developing for Mac systems. During development, thorough testing and adjustments are necessary to ensure compatibility. It is important to adapt the design and functionality for a smooth user experience across different platforms. Staying updated with technology standards is crucial to ensure ongoing compatibility and optimal performance. | The system provides flexibility for clients and developers to view updates from any location. However, it may be slightly more challenging to implement compared to other devices. |
| **Development Tools** | When you use a Mac, you can run Swift, which is a popular programming language. You can also use helpful tools like Notepad++ to write your code. However, Macs can run many other languages too, such as HTML, CSS, and JavaScript, which are used for creating websites. There are also libraries available to help with the design and functionality of the website. Additionally, you can use general-purpose languages like Java, Python, PHP, and Ruby for various programming tasks on a Mac. | Linux is another operating system that can be used with different programming tools like Visual Studio, Eclipse, and Notepad++. These tools make it easier to write code. Linux can also support a wide range of programming languages and tools. Some of the languages you can use on Linux are HTML, CSS, and JavaScript, which are used for creating websites. There are also libraries available to help with designing and making websites work well. Additionally, you can use general-purpose languages like Java, Python, PHP, and Ruby for various programming tasks on Linux. | Using a Mac is easier than using Linux, but both can run similar programming languages. There are helpful tools like Visual Studio and Eclipse that can be used on both systems. Notepad++ is a simple and easy-to-use tool available for both Mac and Linux. These systems can work with various programming languages, including HTML, CSS, and JavaScript for making websites. There are also libraries that help with creating the look and functionality of websites. Additionally, you can use popular general-purpose languages like Java, Python, PHP, and Ruby for different programming tasks on both Mac and Linux. | You can make lots of apps with Android and Swift on any of the three machines. These languages and software work on all of them. There are different programming languages like HTML, CSS, and JavaScript that you can use to build websites. You can also use libraries to help with making the front part of the website look good and work well. For other programming tasks, you can use general-purpose languages like Java, Python, PHP, and Ruby. So, no matter which machine you have, you can create all sorts of apps and websites using these languages. |

**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**: I suggest using Windows devices because they have more software options and it's easier and cheaper to get started. You don't need a lot of expertise to begin working on projects. Plus, you won't have any trouble finding different programs to write your code in.

**Operating Systems Architectures**: <Windows makes it possible for all Windows-based apps to have a pretty look and use the computer's resources. They can show a Graphical User Interface (GUI) and do other cool things. These apps can also work with graphics, multimedia, messaging, and web services. You can use these services by signing in with your own account or using a special server.

* **Storage Management**: Windows 10 has a cool feature called storage sense that helps you look at and organize your files on the hard drive. It shows you how much space they take up. Another cool thing is that you can choose where to save apps, so it's easier to find them. You can also use the cloud to save your data, just like with other devices. The built-in storage system makes it easy to create and save files for big projects, so you won't lose them or accidentally delete them.
* **Memory Management**: When making this game, you'll need to create a database or library to store many pictures. The memory allocation feature makes it easy to store these pictures in a separate place outside of the default picture folder. This helps you keep your entire project together in a safer area on your computer. It's useful when working with your IDE and opening files from it to create the game.
* **Distributed Systems and Networks**: Since each operating system is different, I looked into ways to publish the game so it can run on all devices. I discovered Develop 4, which is a tool for creating games that can work on different platforms. It's an IDE (Integrated Development Environment) that can be used on any device. Once the game is created, you can easily export it to run on the web, iOS, Android, and other platforms, enabling cross-play between different devices. This helps with handling dependencies and compatibility issues.
* **Security**: Windows includes built-in security protection software, but it's advisable to use additional security measures to safeguard user data and information. However, when considering the protection provided by the operating system itself, Windows comes with pre-installed security features. These features scan for malware, viruses, and security threats in real-time. Since threats constantly evolve, the system automatically updates itself to ensure the system and user information remain secure.