

<Name-of-Software-Application>

# CS 230 Project Software Design Template

Version 1.0

## Table of Contents

[CS 230 Project Software Design Template 1](#_Toc115077317)

[Table of Contents 2](#_Toc115077318)

[Document Revision History 2](#_Toc115077319)

[Executive Summary 3](#_Toc115077320)

[Requirements 3](#_Toc115077321)

[Design Constraints 3](#_Toc115077322)

[System Architecture View 3](#_Toc115077323)

[Domain Model 3](#_Toc115077324)

[Evaluation 4](#_Toc115077325)

[Recommendations 5](#_Toc115077326)

## [Document Revision History](#_grjogdjh5fi8)

| Version | Date | Author | Comments |
| --- | --- | --- | --- |
| 1.0 | <02/16/2025> | <Bharat Sharma> | <Well user-based program with multi-platforms> |

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](" \l "_sbfa50wo7nsh" \t "_blank):

To provide further guidelines, my team and I are tasked with evaluating various platforms' characteristics, advantages, and weaknesses for deploying and running the "Draw it or lose it" game application. My team considers both server-side and client-side ideas and the development tools required for each platform; an outline of the evaluation will be included in the software design summary.

## Requirements: The evaluation of Operating platforms for hosting a web-based software application:

The Platform Characteristics:

In this platform characteristics section, our team describes the key characteristics of each operating platform. Our team's focus will be hosting relevant web-based software applications, considering factors such as performance, stability, security, and effective game deployment.

Platform Advantages:

To explain the platform advantages, my team found that each platform hosted the context of the web-based games. For example, Linux offers better scalability and is one of the most important securities. At the same time, other operating systems, such as Window Windows, have higher compatibility with certain or many libraries for software. The new thing will be cross-platform compatibility, an anti-cheat system with strong security, higher performance and optimization with strong networking, and scalable multiplayer users.

Platforms Weaknesses:

To address the platform weaknesses or limitations of each one of the platforms concerning hosting software applications with web based. For instance, Apple provides a smaller user base because Apple is not cost-effective, whereas Windows has numerous licensing and user-based costs. Moreover, other weaknesses can be due to real-time processing, higher graphical demands of games and connecting online, frame rate drop or lag while playing games, and the most important one is hacking or cheating by players on the game with any platforms.

Server-based deployment:

To indicate server-based deployment, each operating platform offers a website hosting method to evaluate how well it can handle accommodating thousands of players. The key components of server-based deployment are the game server, data server, authentication server, match-making server, and content delivery server. These are just a few examples of the most important parts of game applications.

Potential Licensing Costs:

Potential licensing costs provide research and information for every operating system on each platform. Both consider initial costs and ongoing licensing fees, such as subscription-based software like Microsoft Office. Another one will be perpetual licensing, where the user must buy once for multiple uses.

Let's consider the multiple platforms, such as Windows, Linux, Apple, Android, and iOS.

IOS (iPhone Operating Systems):

iOS is known for being secure and user-friendly, with a comfortable environment. It provides higher security features and the best user experience, including Apple Store eco-systems, regular updates, and more substantial developer support.

IOS Pros and Cons

The Apple user has a loyal base, making it an attractive platform for reaching a specific. People think owning a piece of Apple hardware with their strong customer support is luxurious. Their app store goes through strict quality checks, which reduce the risk of malware, or any bugs related to game applications. The many challenges of IOS are limited options, such as Apple computers, iPhones, and iPads, compared to Android eco-systems. IOS has strict guidelines for apps, making distribution challenging for developers.

IOS Server Based Deployment:

It is the same as Android. IOS applications are deployed and distributed through the Apple App Store, where they do not require server-based deployment like some other operating systems, such as Windows and Linux. IOS Developers must be proficient in web development and must have expertise in developing iOS apps using XCode and Swift or its native features. The IDEs for IOS primarily use XCode, which includes robust tools for building web-based developments.

Potential Licensing Costs:

The Apple App Store does not require direct licensing costs; however, it will ask you to pay an Apple Developer program membership fee, which varies from app to app. Apple charges according to clients' needs, such as an Individual Membership fee and an organization membership fee, which is the same: $99 a year. Also, the Apple Store takes commissions on any revenue generated through the app store.

Android OS/ Programing

Android is open source, which Google developed to compete with Apple and Windows, and it is a Linux-based operating system. Android is a mobile-optimized platform known for its significant market share and access to numerous users. Some of the primary key features are Google service integration, multi-device support, open sources, and customizable UI, which are the most relevant and fundamental eco-systems given to Android operating systems, making it one of the most significant market users. The Androids application was developed using Java or Kotlin with the help of HTML, CSS, and JavaScript for web-based development.

Android Pros and Cons

The main Android advantage is to support a massive audience of mobile users, given its dominant market share in the technology industries. The Android market provides more control for developers and a wide range of affordable devices, which creates a variety of hardware choiceless with expandable storage and file management. Android's most common weakness is its fragmentation issues and inconsistent software updates because Android runs on a wide range of devices. It also brings disadvantages, such as Heavy RAM use, security risk, limited customer support, and slow software updates.

Android Server-based Deployment:

Like iOS, Android is distributed through the Google Play Store and other marketplaces, but it must go through the security check provided by Android operating systems. Android does not need server-based deployment, like Windows, Mac, and Linux operating systems. However, Android's official IDE for Android development can be standardized by using Visual Studio

Code or WebStorm.

Android Potential Licensing Costs:

Android does not ask for direct licensing costs, but Google asks for a GMS (Google Mobile Services) fee of $40 per Android phone. Also, Google may charge its developers up to 15-30% commission on revenues. As a game developer, my clients and I have to pay an additional service fee.

Finally, we might see the most similarities between Android and iPhone IOS because they operate the same based on the same markets but different operating systems that are only different. However, the only difference they have is the prices of their products. Let's move on to more of the desktop platforms.

Linux

Linux is an open-source operating system that uses Unix architecture. Linux itself is written in C. Linux's programming language is commonly web-based, which is developed using languages such as Java, JavaScript, Python, and PHP. HTML and CSS are used for the front end. Now, Linux also uses Perl, Ruby, and PHP for web-based applications.

IDEs and Tools

Now, developers use text editors such as Visual Studio Code or Sublime Text for coding because there are many platforms for them, and people are always finding ways to make their jobs easier and more cost-effective. For the development of server-side Linux, use IDEs with PyCharm for Python and JavaScript for WebStorm.

Linux Pros and Cons

We all know by now that Linux is an open and free source easily accessible to anybody, making operating system customization flexible where users modify it. Linux is Lightweight and efficient, with advanced terminal commands ideal for development and systems management. The disadvantages of Linux are that Linux comes with gaming limitations, software compatibility issues, hardware support, fragmentation, and no official support like Windows and Apple.

Impact on Developments

The development teams should bring expertise in Linux-based web development languages and frameworks. If the expertise is familiar with Linux open-source development tools, it may provide with the development process.

Linux Licensing Costs:

For Linux, most of the applications are open sources, which makes most of the time free or may reduce the licensing costs.

Apple/MAC/OS

MAC OS is Apple's most promising operating system, known for its sleek design by Apple engineers. Its user-friendly interface is clean and intuitive, with smooth animation. Apple is like Linux because Mac is also developed using JavaScript, Python, and Ruby for the backend, and HTML and CSS are commonly used for the front end.

IDEs and Tools for MAC

Mac users have imitated the use of coding languages. Mac users use XCode and Visual Studio Code for web development. For their server-side development, they apply tools like PyCharm and WebStorm.

MacOS Development

The Mac platform development is well-suited to the macOS environment familiar to Mac developers. The team should bring Mac-related expertise in web development languages and Mac OS-specific tools like XCode.

Mac Pros and Cons

The Mac is simple and provides higher security, smooth performances, and stability with fewer crashes and longer software support from Apple. If you keep your Mac longer, it will get software updates. However, Macs can be expensive, with limited hardware customization, unlike PCs, fewer gaming supports, and difficulty to repair, and they come with expansive repair costs. Apple produces expansive accessories that support only USB type-C.

Apple Licensing Costs

Most programming languages, like XCode, have licensing fees for certain features. Depending on web development tools on Mac, they may also be free with affordable licensing options.

Windows OS and Programming language

Window and Programing

The widows are one of the most commonly used and bring more users like androids’ users, with an estimated market cap of 75%. Moreover, most new vision windows are Windows 11. Windows is primarily written in C and C++. These are suitable for Windows, but Windows also produces its programming language, such as Visual Studio Code. Windows also allows third-party web developers to run into Windows.

IDEs and Tools

The most used Windows developers are Visual Studio Code and Subline Texting. For Net development, Visual Studio is the best choice. It is developed by Windows and is most compatible with Windows.

The Window Development

The Windows developer is proficient in web development and its programming language, where it may bring expertise in NET applications. Windows has been an essential part of computing for decades, influencing people to work, create, and play with Windows OS.

Windows Pros and Cons

Windows supports various hardware types, from the low to the high end. This makes developers choose Windows for programming and playing games; it supports many games' hardware and performance optimization. 75% of the market is covered by Windows, which makes people easily understand the Windows interface for newer users and the start menu. It has strong cross-platform support tools and cloud integration. However, it brings security vulnerabilities due to cost efficiency and popular choices of people; users must install anti-viruses and regular software updates. It is also less developer-friendly than Linux or Apple.

Windows Licensing Costs

Windows licensing costs, such as Visual Studio, may require further feature access fees, but alternatives are also free. If you Google alternative programming languages, you will find many more, but it is at your own risk.

Summary

In summary, web-based software development tools consist of various programming languages, IDEs, and other tools. Each platform requires specific expertise, with the developer's team needing to be more proficient in multiple languages and frameworks and able to support different platforms at a time. Depending on our needs, we might have to pay licensing fees to access more features. However, some development tools are free and affordable to reduce the financial burden on clients and game development teams.

To evaluate software development clients, multiple supporting are as follows:

**Platform:**

One key consideration when supporting multiple clients is ensuring platform compatibility with numerous cross-platform applications; the clients provide support for desktops, Linux, Mac, Windows, and Android or iOS applications, and they must work seamlessly on multiple cross-platform applications.

Development costs:

Developing costs can vary depending on the number of features available. They also include the effort required to develop and test the application or the effort required by the development teams for cross-platform frameworks such as React Native or Flutter, which could reduce time and costs by allowing code reuse across different platforms.

Time:

Time is influenced by the complexity of the development team and their support of multiple platforms. Each platform may require a unique quirk that developers need to address, potentially leading to longer development timelines.

**Testing:**

The testing application across multiple platforms is crucial to ensure consistent user experiences. It will increase the testing compared with single-platform applications. The application should adapt to different screen sizes and resolutions by providing a smooth experience for desktop and mobile devices. Each platform has its performance gaudiness. The application should be optimized and run smoothly on target platforms, including CPU, RAM, memory, and GPU capabilities. Testing also includes security. Why? Especially in a distrusted environment, the application must design to handle security challenges on different platforms, including protection against vulnerabilities of each platform required. The application architecture should accommodate thousands of players, and as a server-side, the infrastructure should be capable of handling increasing traffic from multiple platforms.  It must launch maintenance and updates according to cross-platform requirements, fix future bugs, and ensure all the platforms receive timely updates without any delays. It must pass all those things listed before lunch the game. Testing game applications and many frameworks will tell us whether or not the game will be a successful launch.

In Conclusions:

The above-mentioned technical requirements on the development team, the platform expertise, cross-platform developments, collaboration, communication, testing efforts, development timelines, resource allocation, and continued maintenance are needed. Due to diverse technical requirements, each team could specialize in their fields to ensure optimized development and problem-solving efficiency and deliver high-quality results from their assigned platforms or teams. On the other hand, some platforms might have to pay fees considering their needs and revenue. For example, apps like Google Play Store and Apple App Store may have fees with membership update options. The development team should consider licensing costs and evaluate the function and requirements of given tools and platforms.