

Draw It or Lose It

# **CS 230 Project Software Design Template**

Version 1.0

## Table of Contents

[**CS 230 Project Software Design Template**](#_l6ti7uoag22u)1

[**Table of Contents**](#_30j0zll)2

[**Document Revision History**](#_grjogdjh5fi8)2

[**Executive Summary**](#_sbfa50wo7nsh)3

[**Design Constraints**](#_2et92p0)3

[**System Architecture View**](#_ilbxbyevv6b6)3

[**Domain Model**](#_8h2ehzxfam4o)3

[**Evaluation**](#_2o15spng8stw)3

[**Recommendations**](#_m8aleynsvzvc)5

## [Document Revision History](#_grjogdjh5fi8)

| Version | Date | Author | Comments |
| --- | --- | --- | --- |
| 1.0 | 10/18/2020 | Joseph Mead |  |

## [Executive Summary](#_sbfa50wo7nsh)

Problem Statement

The Gaming Room would like to develop a web based game that serves multiple platforms based on an Android app. The game, Draw it or Lose it, features a library of stock drawings, which are slowly revealed, while teams try to guess the drawing before time expires.

Software Requirements

* The game is to be web based and cross platform.
* Ability for multiple teams to play in one game.
* Unique identifiers must be assigned to each Game, each Team, and each Player.
* Ability to check existing Game, Team, and Player names.
* Only one game can exist at a time

## [Design Constraints](#_2et92p0)

* Presumably, the game will be run within a web browser.
* A server will be required to host the game.
* Compatibility with different web browsers running on different operating systems is required.
* Software must be cross-platform

## [System Architecture View](#_ilbxbyevv6b6)

## [Domain Model](#_8h2ehzxfam4o)

The Game, Team, and Player classes all inherit attributes from the Entity class, which is an example of Inheritance. The Entity class features Polymorphism by containing overloaded constructors to create and entity object. The Entity class contains the private properties of id and name, which are can be accessed but not changed, which shows Encapsulation. Likewise, the teams and players are private properties of the Game and Team classes respectively. The GameService class can have 0 to many associations of the Game class, which can have 0 to many associations of the Team class, which can have 0 to many associations of the Player class. All classes are separate and reusable, along with SingetonTester class, which checks that only one instance of the GameService class can take up memory at a time. These classes are all reusable therefore showing Portability.

****

## [Evaluation](#_2o15spng8stw)

| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| --- | --- | --- | --- | --- |
| **Server Side** | In 2018, Apple discontinued many features for their MacOS Server, including the ability to host websites. The new version appears to focus on device management and file storage for home use. This makes it a poor choice for hosting the game.  If one absolutely wanted to use a macOS Server, there may be some work-arounds by using third party software. This negates the biggest selling point of the macOS Server, which is that it was the easiest to use and didn’t require as much training to manage. (Vaughan-Nichols, 2018) | Linux is an open source operating system and IT infrastructure platform. Because it’s open source, the software is free. Linux is generally more secure than Windows. Linux usually requires less computer resources to run than Windows.  Linux features a tree directory which makes it difficult for a hacker to install software with root access on the server. The fact that it’s open source is also beneficial for security. Technicians are able to see the code and tend to identify and report holes in security much faster than with propriety software.  Linux can be more challenging to setup and may require someone with expertise to set it up and maintain it.  If one wanted to implement LDAP protocol, there are a variety of options. The most popular option is OpenLDAP.  You can also run a Virtual Machine on Linux, or, for programs that can be run within containers, Docker runs natively on Linux.  Linux is cloud capable. The advantage of using the cloud is that one doesn’t have to constantly upgrade hardware for a growing user base. The disadvantage is that there is a monthly fee for it’s use.    Linux is eminently scalable and features server based Deployment. It’s likely the cheapest option available. A potential issue is the availability of a skilled technician to install and maintain it. (Noyes, 2010) | Windows is designed by Microsoft. In order to use it, one must buy the operating system and may be required to pay a yearly fee.  Windows servers are easier to set up and manage. Microsoft also offers customer service in the event that the technician operating the server can’t resolve an issue.  Windows Servers are very secure, but they are often targeted by hackers due to the popularity of the platform. Programs that require elevated permissions are stored separately in %SystemRoot%. Programs attempting access they don’t have trigger an UAC prompt which is helpful in preventing unauthorized access.  Windows servers use Microsoft Active Directory for authentications which uses LDAP as a basis.  Virtual Machine and Docker both run on Windows Server.  Window Servers are cloud capable.  With the availability of support and the ease of set up and implement, Microsoft Server is a good choice for less skilled technicians. (Microsoft, 2019) | Surprisingly, it is possible to install a web server on an android device, but it’s only value would be for testing. It would not be capable of handling any kind of volume. |
| **Client Side** | MacOS is the second most popular desktop operating system in the world. From 2013 to 2019, the MacOS market share grew from 8% to 17% (Statcounter, 2002). The two most popular browsers on MacOS are Chrome and Safari. Chrome and Safari combined make up around 85% of all browser usage on any system (StatCounter, 2020). Compatibility on those two browsers is essential. Neither Safari nor Chrome are supporting Java Applets. If the game is to be written in Java, there are some Java compilers for the web that convert it to HTML5, WebAssemby, or JavaScript, all of which are supported by Safari and Chrome. | The Linux market share grew from 0.9% in 2013 to 1.9% in 2019 (Statcounter, 2002).  The most popular browser on Linux is Chrome (StatCounter, 2020). There shouldn’t be a huge difference in how the site behaves on Chrome regardless of the operating system. However, screen resolution could have an impact.  In addition to Chrome, there are a variety of other popular browsers on Linux. FireFox, Oprah, Vivaldi are also popular options. These browsers also do not support java applets. Java may be used, but it should be compiled to convert it to HTML5, WebAssemby, or JavaScript, all are supported by most browsers on Linux. | Windows is the most popular OS in the world with a market share of 77% (down from 90% in 2013) (Statcounter, 2002). Chrome is also the most popular browser on Windows. There should not be a huge difference in the performance on Chrome across different OSes. While surprisingly not popular, the default browser in Windows is Edge, which like Chrome, does not support applets. The one browser that does continue to support applets is on Windows, Internet Explorer. However, Internet Explorer is only used by 2.5% of desktop users (StatCounter, 2020). The majority of browsers used on Windows support HTML5, WebAssemby, or JavaScript. | Safari and Chrome are the most popular browsers on mobile devices (StatCounter, 2020). The most important consideration for an HTML based browser game on mobile systems is the different resolutions of the devices. There are tools that assist with testing of sites on different browsers on different operating systems that can assist with development. |
| **Development Tools** | Apple’s programming language is Swift and the IDE is Xcode, but Apple has support for most programming languages and there are a variety of IDEs available including Eclipse. Java is not installed by default and takes a little effort to implement (Apple, 2020). | Linux has been home to just about every programming language over the last 24 years. Linux is Unix based so C is sometimes thought of as the language of choice but, C++, Python, Java, C#, Fortran, Pascal, COBOL, Lisp and more have all been featured in Linux (Bolton, 2015). There are many IDEs available for Linux. A couple of popular options are Eclipse and Netbeans. | Windows is mostly written in C and C++, but they have announced recently that they will be switching to Rust in the future. This is due to vulnerabilities that continue to come up with C and C++ (Tung, 2020). Visual Basic is a Microsoft language that is used for many GUI applications. Microsoft’s IDE is Visual Studio, but many IDE’s are available on Windows including Eclipse. | While it’s likely that there are IDEs available for mobile devices, it’s uncommon for anyone to use a mobile device as a development tool. |

## Recommendations

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

1. **Operating Platform**:

As outlined above, there are only two realistic options to choose from: A Linux based server or a Windows server. Between the two, I recommend a Linux server. While both operating systems are scalable, secure, and cloud capable, Linux is generally considered more secure, less resource hungry, and most importantly, it’s usually free. It is important to point out that there are many different flavors of Linux and not all are free. Red Hat, for example, requires a subscription. If one wanted something similar to Red Hat but didn’t want to subscribe, there is CentOS, which is a derivative of Red Hat and is used on IBM servers. (Kili, 2020) My recommendation would be an Ubuntu server because it is the most popular, contains supports big data, visualization, containers, Internet Of Things, and is compatible with most common public clouds. Ubuntu also features a subscription support service that can assist the management.

1. **Operating Systems Architecture**:

Linux has a monolithic kernel which means that user services and kernel services are implemented under the same address (as opposed to a microkernel architecture that has different address spaces for user and kernel services). (akash1295, 2019) The advantages of this architecture is that commands are executed faster. The disadvantage is that if one service crashes, the whole system will crash. Windows architecture is called a hybrid kernel. It’s an attempt to combine aspects of both monolithic and microkernel architectures. It contains the modular structure of the microkernel by running kernel code as servers in the user space, but most operating system services are still in kernel space. (Techopidia, n.d.) There are many who claim that the hybrid architecture isn’t significantly different than the monolithic kernel. Linus Torvalds claims it’s nothing more than a marketing ploy. (Mikov, 2006) Given the similarities, the architecture of each operating system didn’t factor into the recommendation of Linux over Windows.

1. **Storage Management**:

There are two options for storing the data. One could store the data on drives in house or one could rent space on a cloud server. In house storage requires a larger initial investment, but may be cheaper in the long run as it avoids a monthly payment. Other advantages include keeping third parties away from accessing the data, and the ability to access the data even when the internet is down. The cloud server gives one the ability to add storage without having to purchase more hardware and the ability to keep the data intact in the event of a disaster at the local site. I recommend starting with local drives because, for smaller to medium size businesses, it tends to be less expensive. In the event that the game becomes larger, then it may be advantageous to switch to a cloud server. (SysGen, n.d.)

1. **Memory Management**:

Linux uses RAM as its main memory. It makes use of virtual memory in the event that RAM becomes low. This is where space on the drive is used to store data. The process of storing and retrieving data from disk is called paging. (ApacheBooster, 2018)

1. **Distributed Systems and Networks**:

In order to deliver the game to users who may be geographically far apart, distributed systems can be employed to reduce latency. This is where there are multiple servers in multiple locations. The other advantage of this is that in the event of an outage in one location, there are other locations that may still be in service which will lessen the impact. The disadvantage of this system is that it’s much more complex than a single client-server system. Synchronizing the changes to data can be a challenge in the event of network failures between servers. (Confluent, n.d.)

1. **Security**:

One of the advantages that a Linux server has over Windows is the file structure makes it difficult for a user to get root access even if they were able to achieve unauthorized access. In addition, there are a number of other strategies that can be employed to increase security. One important measure is to have the data backed up. One way to do this is using a cloud server, which is outlined in the storage management portion of this document. A back-up for a local drive can be achieved as well. Another effective security measure is the use of a firewall. This is where one can apply rules in iptables that filter incoming, outgoing, and forwarding packets. Other measure that can be taken include strong password enforcement, disabling of IPv6 (because of the vulnerabilities it would introduce and the fact that it isn’t necessary on a server), audit of network ports and disabling ones that are unwanted, and the use of a secure shell that encrypts communications with the server. (TecMint, 2015)

# References

(n.d.). Retrieved from https://www.tecmint.com/10-best-linux-server-distributions/

akash1295. (2019, August 16). *Monolithic Kernel and Key Differences from Microkernel*. Retrieved from GeeksForGeeks: https://www.geeksforgeeks.org/monolithic-kernel-and-key-differences-from-microkernel/

ApacheBooster. (2018, March 19). *Types of Memory Management Techniques To Manage Server Performance*. Retrieved from Apachebooster.com: https://apachebooster.com/blog/types-of-memory-management-techniques-to-manage-server-performance/

Apple. (2020). *Swift*. Retrieved from Apple.com: https://www.apple.com/swift/

Bolton, D. (2015, March 9). *Best Programming Langues for Linux Devs*. Retrieved from Dice.com: https://insights.dice.com/2015/03/09/best-programming-languages-linux-devs/

Confluent. (n.d.). *Distributed Systems - The Complete Guide*. Retrieved from Conflent: https://www.confluent.io/learn/distributed-systems/

Kili, A. (2020, July 27). *10 Best Linux Server Distributions of 2020*. Retrieved from TecMint: https://www.tecmint.com/10-best-linux-server-distributions/

Microsoft. (2019). *The Ultimate Guide to Windows Server 2019*. Retrieved from Microsoft.com: https://info.microsoft.com/ww-landing-ultimate-guide-to-windows-server-2019.html

Mikov, T. (2006, May 8). *Hybrid (micro)kernels*. Retrieved from Real World Technologies: https://www.realworldtech.com/forum/?threadid=65936&curpostid=65915

Noyes, K. (2010, Aungust 3). *Why Linux Is More Secure Than Windows*. Retrieved from PCWorld.com: https://www.pcworld.com/article/202452/why\_linux\_is\_more\_secure\_than\_windows.html

Statcounter. (2002, September). *Operating System Market Share*. Retrieved from GS.StatCounter.com: https://gs.statcounter.com/os-market-share

StatCounter. (2020, September). *Browser Market Share Worldwide*. Retrieved from GS.StatCounter.com: https://gs.statcounter.com/browser-market-share

SysGen. (n.d.). *The pros and cons of Cloud vs. in-jouse Servers*. Retrieved from SysGen: https://sysgen.ca/cloud-vs-in-house-servers/

Techopidia. (n.d.). *Hybrid Kernel*. Retrieved from Techopedia: https://www.techopedia.com/definition/27004/hybrid-kernel

TecMint. (2015, January 3). *25 Hardening Security Tips for Linux Servers*. Retrieved from TecMint.com: https://www.tecmint.com/linux-server-hardening-security-tips/

Tung, L. (2020, May 1). *Microsoft: Our Rust programming language Windos runtime library is now in preview*. Retrieved from ZDNet.com: https://www.zdnet.com/article/microsoft-our-rust-programming-language-windows-runtime-library-is-now-in-preview/

Vaughan-Nichols, S. J. (2018, January 29). *The death of macOS Server*. Retrieved from ZDNet.com: https://www.zdnet.com/article/the-death-of-macos-server/