

# Draw It or Lose It

# **CS 230 Project Three 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 | 12/20/2020 | Amber Foster | The Final Evaluation |

**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](#_sbfa50wo7nsh)

In this program, we are expanding a web-based game application that can run on different operating system platforms with different software patterns. The game currently works on android, but we need it to be able to work on Windows, Linux, and Mac OS. In order to find the best platform to run the gaming program on, we need to evaluate the different operating systems that are being considered.

## [Design Constraints](#_2et92p0)

The design constraints that we initiated an online multiplayer game that plays on the top three major operating system platforms. The client wants the user authentication methods implemented into the application so that the user can access the data that remain secure from malicious activity. We already have it on the android and now need to work the game app into other mobile devices like Windows, Linux, and Apple. So, we need to have one or multiple teams that can be involved in this project. Also, only one case of the game can exist at any time and needs to run on many platforms.

## [System Architecture View](#_ilbxbyevv6b6)

Please note: There is nothing required here for these projects, but this section serves as a reminder that describing the system and subsystem 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](#_8h2ehzxfam4o)

In the UML class diagram that was given, we can see that the game “Draw It or Lose It” relats to the game that applies into “The Gaming Room” application. The game will have multiple teams with multiple players in them. The game will consist of two teams that will be played against each other. The game will consist of a picture where one player will draw the picture and the other player will guess what the picture is within the time limit of 30 seconds. If the time runs out, then that team lose, but if they win then they will receive a point for their team. For the team that has the most points at the end will win the whole game.

**Diagram

Description automatically generated**

## [Evaluation](#_2o15spng8stw)

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 easy accessibility and server configuration.  Flexible terminal commands to design the server, access and make changes.  Easy to use graphical user interface. | Difficulty navigating the platform.  Linux is cost friendly by making changes.  Mac and Linux have a command shell for simple server configuration and accessibility. | Windows server side has more available software compared to other Mac and Linux systems.  Windows server are user friendly GUI and has Command prompt. | Mobile device specifications vary from user to user.  Specifications are better in other devices. |
| **Client Side** | **EXPENSIVE:** Mac has a moderate range of expensive option for their users.  **COST:** Mac are the same in cost as the Window system.  **TIME:** Mac users need moderate time and skills to support a Mac setup. | **EXPENSIVE:** Linux has a maximum range of expensive option for their users.  **COST:** Linux are less expensive in cost for Mac or Window system.  **TIME:** Linux users need maximum time and skills to support a Linux setup. | **EXPENSIVE:** Windows has a minimum range of expensive option for their users.  **COST:** Windows are more expensive than the Linux systems.  **TIME:** Windows Users need minimum time and skills to support a Windows setup. | **EXPENSIVE:** Mobile platforms are hard to perform applications created for other platforms. Provides flexibility to clients or even developers to see updates at any place.  **COST:** Mobile platforms are less expensive than the other systems.  **TIME:** Mobile platforms users need maximum time and skills to support a mobile device setup. |
| **Development Tools** | Development  tools for Mac systems include  PyCharm, Eclipse, Visual Studio, GitHub, Notepad++ and online developing tools. | Development tools for Linux systems include GitHub, Visual Studio, PyCharm, Eclipse and Notepad++. | Developer tools for Windows systems include Eclipse, command prompt, PyCharm, Eclipse, Visual Studio, GitHub and Notepad++. | Developer tools for Mobile platforms include GitHub, Visual Studio, command prompt, PyCharm, Eclipse and Notepad++. |

**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**: There is one option that is the best option that stands out as an operating platform that will allow The Gaming Room to expand Draw It or Lose It to other computing environments. Linux server is the best option for installing and setting up the application that is on a web server. The cost, stability, and scalability are all benefits of this platform. Linux stands out among other operating systems when it comes to the cost, scalability, flexibility, stability, reliability, security of ownership. After installing and setting up the application on the Linux server, it will allow us to operate on multiple client operating systems.
* **Operating Systems Architectures**: First is the hardware, which is needed when dealing with computer architecture. Second is the kernel, which is needed to implement the memory that is set apartfrom applications. Third is the application, which is the user application that is ran within the kernel. The architecture of a Linux system that starts out with the kernel, then goes to run the CPU, memory, and other hardware resources. As for the tools that are used with the operating system when the applications are used to help assist the kernel.

* **Storage Management**: Cloud servers and databases are very useful and can be more cost effective when using a Linux server that is set up to a local database that has a file system using Bash. When it comes to instantiating multiple game services on the cloud that are spread out over several systems making for an even better stability.
* **Memory Management**: Linux uses virtual memory that use swap space, which is a mixture of both the RAM and the disk space, and the kernel helps with keeping the memory separated from the other additional processes. Windows uses dynamic memory that use linking, which is a development to helps share the limited memory resources. Both Linux and Windows operating systems develop a categorized file system that can gain the access to secure authorizations on the operating system.
* **Distributed Systems and Networks**: Local servers have a big risk of running into power outages and interruptions while running Linux. Cloud services can have issues among multiple servers and their locations. But both the local servers and the cloud services can increase their reliability to the client. The Client-server architecture is the best way when trying to avoid the outages and it guarantees quantity for the interconnection of platforms, systems, and applications when using cloud services. However, cloud migration can also have more resources.
* **Security**: Linux has a great operating system that can be used as a cloud service or a single server which could be alarming for Windows systems by having huge security risks. A huge advantage that Linux has is that they are supported by numerous individuals and big companies that are invested which helps test all their inventory from Linux. The firewall and security that helps protect the cloud structure itself for many of the cloud services that are utilizing in real-time. Linux also has many stabilities for their users, like for example how the users can choose to update or not to new security features at any time. For other users that never shutdown their Linux systems, Linux has the reputation for low crashing reports if the users are running without having to restart the system.

**Explain the operating system architectures for different operating systems:**

Monolithic operating system is an operating system architecture where the entire operating system is working in kernel space and is alone in supervisor mode. Every component of the operating system is contained in the kernel and can directly communicate with others. The kernel typically executes with unrestricted access to the computer system.

Layered Operating System is an operating system that became larger and more complex, purely monolithic designs became unwieldy. The layered approach to operating system attempts to address this issue by grouping components that perform similar functions into layers. Each layer communicates exclusively with those immediately above and below it. Lower-level layers provide services to higher-level ones using an interface that hides their implementation.

Microkernel operating system components such as process management, networking, file system interaction and device management execute outside the kernel with a lower privilege level. Microkernels exhibit a high degree of modularity, making them extensible, portable, and scalable. Further, because the microkernel does not rely on each component to execute, one or more components can fail, without causing the operating system to fail.

Networked operating system enables its process to access resources that reside on other independent computers on a network. The structure of many networked and distributed operating systems is often based on the client/server model. The client computers in such a network request resources such as files and processor time via the appropriate network protocol. The servers respond with the appropriate resources.

**Analyze the differential file system components that represent a collection of data:**

A file is a named collection of related information that is recorded on secondary storage such as magnetic disks, magnetic tapes and optical disks. In general, a file is a sequence of bits, bytes, lines or records whose meaning is defined by the file’s creator and user. File Structure types also can be used to indicate the internal structure of the file. Internal file structure, locating an offset within a file can be complicated for the operating system. Disk systems typically have a well-defined block size determined by the size of a sector.

**Explain the functions of memory and storage management:**

Memory management refers to the different methods and operations that are responsible for obtaining maximum memory utility, organizing the processes and programs that run in such a way that the available space is best exploited. The current methods are based on the memory management that was used in the past. **Linking instructions and data to memory** can be done in three stadiums which are build time, load time, runtime and overlays.

**For Identify memory management techniques:**

There are three major memory management techniques are swapping, paging and segmentation.

**Describe distributed systems and the networks that interconnect them**

A distributed system is defined as a collection of autonomous computers connected by a network, and with the appropriate distributed software for the system to be viewed by users as a single entity capable of providing computing facilities. Distributed systems are deployed on a variety of hardware platforms, from a few workstations connected by a local area network, to the Internet, a collection of interconnected local area and wide area networks, which fit millions of computers. Distributed system applications range from the provision of computing capacity to groups of users, to banking systems, multimedia communications, and span virtually all commercial and technical applications of computers. The requirements of such applications include a high level of reliability, security against external interference and privacy of the information that the system maintains.

There are numerous examples of distributed systems and networks used in everyday life in a variety of applications. Examples of distributed systems, which are the World Wide Web, network file server, banking network, redes peer-to-peer, process control systems, sensor networks, grid computing. For other common operations include the following a file pointer, file-open count, disk location of the file and access rights.

**THE REFERENCES**

Silberschatz, A. (2010). Wiley. Operating system concepts. (BOOK).

Lavieri, Edward. (2019). Packt. Hands-On Design Patterns with Java. (BOOK).

Cesati, Marco; Bovet, Daniel P. (2005). O'Reilly. Understanding the Linux Kernel, 3rd Edition (BOOK).