

Draw It or Lose It

# **CS 230 Project Software Design Template**

Version 3.0

## 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 |
| --- | --- | --- | --- |
| 3.0 | 04/15/2023 | Linda Nguyen | Addition of operating systems recommendations |

**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)

The Gaming Room would like to develop a web-based game that is based on their current game, Draw It or Lose It. The current version of Draw It or Lose It is only available on Android. The Gaming Room would like to seek a solution to serve multiple platforms.

## Requirements

Each game will have the ability to have one or more teams involved. Each team will have multiple players assigned to it. Game and team names must be unique to allow users to check whether a name is in use when choose a team name. Only one instance of the game can exist in memory at any given time.

## [Design Constraints](#_2et92p0)

The project will need to use a platform-independent programming language such as Java to accommodate for multiple platforms. The game will need to be adjusted to the hardware requirements of each platform. What is the estimated cost of this project? Will The Gaming Room provide the assets or will CTS be expected to create them?

## [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 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](#_8h2ehzxfam4o)

The Entity Class is the base class which contains “name” and “id.” The Game Class, Team Class, and Player Class inherit all the attributes and methods from Entity class. The GameService Class, the Game Class, the Team Class, and Player Class are associated with each other. A GameService can have 0 to many Games. Game can have 0 to many Teams, and Teams can have 0 to many Players. The attributes and methods that cannot be accessed by another are represented by “-“.

**"The Gaming Room UML diagram. The top of the diagram is labeled as com dot gamingroom. Test boxes are placed in two layers. The first layer has three text boxes and the second layer has four of them. In the first layer, the 'ProgramDriver' textbox points to 'SingletonTester' textbox. The 'ProgramDriver' textbox contains the text 'asterisk main round brackets.' The 'SingletonTester' textbox contains the text 'asterisk testSingleton round brackets.' The arrow between these two text boxes are labeled 'open two angle brackets uses close two angle brackets'. In the second layer, there are 'GameService', 'Game', 'Team', and 'Player' text boxes. The 'GameService' textbox has texts arranged in two layers. The first layer contains games colon List open angle bracket Game close angle bracket, nextGamesId colon long, nextPlayer Id colon long, nextTeamId colon long, and service colon GameService. The second layer contains GameService round brackets, getinstance round brackets colon GameService, addGame open parenthesis name colon String close parenthesis colon Game, getGame open parenthesis id colon long close open parenthesis colon Game, getGame open open parenthesis name colon String close open parenthesis colon Game, getGameCount round brackets colon int, getNextPlayerID round brackets colon long, and getNextTeamId round brackets colon long. The 'GameService' box is connected with the 'Game' textbox with a line labeled 'zero dot dt dot asterisk'.  The 'Game' textbox also contains text in two layers. The first layers contains the text teams colon List open angle bracket Team close angle bracket. The second layer has Game open round bracket id colon long comma name colon String close parenthesis, addTeam open parenthesis name colon String close parenthesis Team, toString round brackets colon String. The 'Game' textbox is connected with the 'Team' textbox with a line labeled 'zero dot dt dot asterisk'. The 'Team' textbox also contains text in two layers. The first layers contains the text players colon List open angle bracket Player close angle bracket. The second layer has Team open parenthesis id colon long comma name colon String close parenthesis, addPlayer open parenthesis name colon String close parenthesis colon Player, and toString round brackets colon String. The 'Team' textbox is connected with the 'Player' textbox with a line labeled 'zero dot dt dot asterisk'. It contains the text Player open parenthesis id colon long comma name colon String close parenthesis and toString round brackets colon String. The 'Game', the 'Team, and the 'Player' boxes point to the 'Entity' textbox in first layer. The 'Entity' textbox contains text in two layers. The first layer has the text id colon long and name colon String. The second layer has Entity round brackets, Entity open parenthesis id colon long comma name colon String close parenthesis, getId round brackets colon long, getName round brackets colon String, toString round brackets colon String.**

## [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 OS has a Apache, a popular web server. Apache has extensions for deploying Java applications and is UNIX-based. MySQL database is included. Can generate dynamic server side-content with GIs and Java Servlets. Integrated OpenSSL provides support for strong 128-bit encryption. May have performance issues when traffic heavy. Customizable code can mean new bugs and errors. | Rarely experiences cyber threats and malware. Linux Hosting service licenses are free. System is customizable and supports cooperative work. Not all versions have long-term support. Does not use a GUI and requires some experience to operate. There is no registry so will not slow down. There is limited hardware support. | Utilizes Web Info Services to host websites. Microsoft charges licensing to businesses and others who need to use Windows Hosting. Costs can add up when installing SQL and Exchange Servers. Windows servers have multiple sockets that allow for hosting multiple websites simultaneously. | Mobile cloud hosting is scalable, so companies can pay for what they use. There is no need for server hardware on a cloud. Backup is more reliable. Security may be an issue because data is accessible from any device with an internet connection. |
| **Client Side** | Apple products are generally more expensive and ship with fixed hardware specifications. Most popular web browsers run on Mac OS. Most popular applications run are supported on Mac OS. Mac can build both Android and iOS apps.  Users need a moderate amount of expertise. | More privacy for users.  Is open-source and free, but can be difficult for some users because it is necessary to know commands. Uses little disk space, memory, and processes. Can handle traffic because it keeps more of the system resource available for the site. Users need more experience to use Linux. | Windows is scalable to accommodate spikes in customer traffic or to scale memory and other resources. It is familiar to most users and has an easy to use interface. There is integration with hosted apps and services.  Users need a moderate amount of expertise. | Costs more to develop than web apps and requires regular updates for bugs and security issues. There are bandwidth limitations to prevent clients from overburdening the system. Due to large amounts of storage space, data management can be an issue.  Users need a moderate to high amount of expertise because mobile devices may be more difficult to implement. |
| **Development Tools** | Perl, Ruby, and Python, and other common languages such as Java. | Python, C++, C, Perl, Java. | Windows web hosting uses the .Net Framework ASP, Visual Basic Language, Python, Perl, REXX | Swift, PHP, C#, Objective-C, Java, Kotlin, Scala, Python, C++ |

## 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**: The recommended operation system is a Windows operating system because it is run on a variety of different devices such as a phone, tablet, and PC. It works with other operating systems and devices. It also has a variety of established software for developers such as Azure and Visual Studio.
2. **Operating Systems Architectures**: Windows operating systems consist of a user mode and kernel mode. There are three main environment subsystems which are the Win32 subsystem, OS/2 subsystem, and a POSIX subsystem.
3. **Storage Management**: Azure storage is an appropriate management system for Draw It or Lose it. It is durable and highly available, secure, scalable, and accessible. It is managed, meaning hardware maintenance, updates, and other issues are managed automatically for the user. Azure Files offers fully managed files in the could that are accessible through SMB and NFS protocols, and Azure Files REST API. SMB Azure files shares are accessible from other platforms such as Linux and macOS clients and can be cached on Windows servers with Azure File Sync. This would allow for fast access in proximity to where the data is being used. NFS Azure file shares are accessible from Linux Clients. Azure Blob storage is also an option as it is designed for serving images or documents directly to a browser.
4. **Memory Management**: Windows has a kernal-mode memory manager component that manages physical memory. It is in the form of RAM. The memory manager manages the allocation and deallocation of memory virtually and dynamically. It supports the concepts of memory-mapped files, shared memory, and copy-on-write. Kernel-mode drivers allocate memory for storing internal data, buffering data during I/O operations, and sharing memory with other kernel-mode and user-mode components.
5. **Distributed Systems and Networks**: Windows has the Xbox platform that can help Draw It or Lose expand to gaming consoles and PC gaming. As an Xbox game Draw It or Lose it can utilize Project xCloud to expand and run on any device and is powered by Microsoft Azure. With Azure Gaming, the game can be built and scale across 54 global regions, including China accordind to Microsoft’s website. Project xCloud is currently limited to a set of data centers, but more are being developed.
6. **Security**: Windows is secure because a website will be redirected through the use of the server, ensuring safety of the site and data. Windows automatically updates user’s systems to ensure that they are protected against the latest threats and vulnerabilities. Windows can protect between various platforms with Windows Defender Remote Credential Guard. It protects the user’s credentials over a remote desktop connection by redirecting Kerberos requests back to the device that is requesting the connection and provides single sign-on sessions.