

Draw It or Lose It

# **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 | 03/18/2023 | Brent Longstreet | Updated summary, design constraints, and domain model. |
| 1.1 | 04/01/2023 | Brent Longstreet | Updated evaluation section. |
| 1.2 | 04/15/2023 | Brent Longstreet | Updated recommendations section. |

## [Executive Summary](#_sbfa50wo7nsh)

The Gaming Room wants to develop a multi-platform supported web-based game. The name of the game is “Draw It or Lose It” and is currently available as an Android app only. The game consists of four rounds that last one minute each. The drawings are pulled from a large library of stock drawings as clues. The current team will guess until the timer runs out. If the current team doesn’t correctly guess the puzzle in the allotted time frame, the remaining teams are allowed one guess each to solve the puzzle with a 15-second time limit.

## Requirements

## [Design Constraints](#_2et92p0)

* One or more teams are needed
* Multiple people per team
* Game and Team names must be unique to allow users to check whether the name is in use or available
* Only one instance of the game can exist in memory at any given time
* Must run on multiple platforms

## [System Architecture View](#_ilbxbyevv6b6)

## [Domain Model](#_8h2ehzxfam4o)

Player, Game, and Team classes all have “is a” relationship with Entity class. This means that all three classes is a Entity. Additionally, all three classes are inherited from Entity. In the UML diagram it can be represented as inheritance. The relationship between Team and Player is of “has a” relationship. Team has Players, Game has Teams, and GameService has Games. In the UML diagram it can be represented by aggregation. A “has a” relationship means that an instance of a class has a reference to an instance of another class. The diagram depicts that GameService has a reference of Game, Game has a reference of Team, Team has a reference of Player. A single GameService can have multiple Games. A single Game can have multiple Teams. A single Team can have multiple Players.

**"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)

Each of the operating platforms offer a server-based deployment method where the website will be hosted.

| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| --- | --- | --- | --- | --- |
| **Server Side** | Advantages:  MacOS security features such as built-in firewall and encryption capabilities.  MacOS is known for its compatibility with industry-leading software and applications.  MacOS offers excellent development tools such as Xcode/Swift.  Disadvantages:  MacOS can only run on Apple hardware.  Apple hardware tends to be the more expensive option.  Finding a hosting provider to host a web-based application will be more difficult since it is not the most popular. | Advantages:  Linux is known for its stability.  Linux is an open-source operating system.  Linux is free and can run on a wide variety of hardware.  Disadvantages:  Linux’s user interface isn’t known to be user-friendly.  Linux is more complex than other popular operating systems.  Linux can run into some compatibility issues with software and applications. | Advantages:  Windows is the most well-known, and people/developers are more comfortable.  Windows is compatible with a wide variety range of software and applications.  Windows offers excellent dev tools (VS studio).  Disadvantages:  Windows isn’t cost-effective.  Windows is known for being resource-intensive compared to other operating systems. | Advantages:  Mobile devices are easily transportable and accessible.  Mobile devices offer the use of touch screens.  Mobile devices are often cheaper than the cost of a desktop computers.  Disadvantages:  Mobile devices typically have significantly less processing power than that of a desktop computer.  Mobile Devices have a fraction of the storage space of that of a desktop computer.  Mobile devices are less secure due to being easier to lose. |
| **Client Side** | MacOS will require an individual to use apple hardware to develop in Mac. The hardware is typically not cost effective. MacOS will require a developer with expertise in swift. | Linux will require an individual that has experience with Python. Linux will cost the most in development time due to its complexity. | Windows will require the most in expertise. The developer will need to be experienced with .NET. | Mobile devices will require a specialized developer for mobile applications. The developer will need to worry about user interaction as touch screen and layout are completely different from web application. |
| **Development Tools** | A developer will require a Mac Book. They will need to develop the application using Swift programming language. The IDE’s used will typically be VScode or Xcode. | The IDE most commonly used IDE in Linux is Eclipse. The developer typically writes the program in C/Python. | Windows most relevant IDE is Visual Studio/ Visual Studio Code. The application will be written in C++/C#. | Mobile devices development process differ depending on which operating system the software us built for. The IDE for Android is Android Studio and is written in Java/Kotlin. The IDE for iOS is Xcode and is written in Swift. |

## 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 operator platform I would recommend to The Gaming Room would be Windows. The selection was decided due to Windows having the following benefits:

* Windows is the most popular operating system.
* The platform is a popular choice for game development.
* Windows is developer friendly with built-in IDE’s such as Visual Studio and Visual Studio Code.
* The operating system offers a wide range of compatible software for development.
* Windows allows for applications to be scalable and secure.

1. **Operating Systems Architectures**: The architecture of Windows operating system is broken down into layers. The kernel layer is known for low-level operating systems services like memory management and process scheduling. The executive layer provides higher-level operating system services like network protocols and security. The user mode layer includes the user interface and applications.
2. **Storage Management**: An appropriate storage management system to be used with the Windows operating system is NTFS (New Technology File System). NTFS is the default file system Windows uses. NTFS offers support for files sizes up to 16 exabytes and volumes up to 256 terabytes. In addition to its scalability, NTFS has a built-in file compression feature.
3. **Memory Management**: Windows uses several different memory management techniques for memory optimization. The platform uses virtual memory to allocate memory resources and allow for the disk space to be used as an extension of physical memory. Additionally, Windows uses the memory management technique known as paging. Paging allows for Windows to transfer pages of memory between the physical memory and disk and prioritize more active processes.
4. **Distributed Systems and Networks**: Having Draw It or Lose It being able to communicate between various platforms is essential to the game’s design. Key steps will need to take place for this to be possible. Choosing the correct network communication protocol is important. The difference between TCP and UDP can be huge for a game’s performance. This decision can be made based on the games security and latency requirements. Additionally, a backend will need to be implemented in order to be able to handle these requests. Popular cloud computing services, such as AWS (Amazon Web Services) and Microsoft Azure are excellent choices. Depending on how many concurrent games and the number of requests will decide which service is best for the game’s requirements.
5. **Security**: Security and protecting the client’s information is a huge priority. Windows provides some excellent options in order to protect the user’s information between different platforms. Windows has numerous encryption technologies, such as BitLocker and SSL (Secure Sockets Layer) in order to protect user data. The operating system also offers MFA (Multi-Factor Authentication). Features such as biometric and mobile device authentication can help ensure the intended user is accessing the information. Another layer of protection is Windows built-in antivirus and firewall software to provide protection against malicious content.