

<Name-of-Software-Application>

# **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 | <02/18/22> | Sarah Brady | Final draft of added recommendations related to the design of the Draw It or Lose It gaming software. |

**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 company, The Gaming Room is looking to develop a web-based game based on the popular Android game, Draw it or Lose that they would like to develop. With this, it needs to able to serve on multiple platforms and would be similar to the 1980’s game show, Win, Lose or Draw.

## [Design Constraints](#_2et92p0)

* Must run on multiple platforms (Android, IOS and other web platforms)
* Must have unique Team names and game names and have ability to check names
* Must be able to have multiple players on teams
* Ability to alert lead team player if a team name already exists
* Unique ID’s must be used for instance of player, team and the game

Above are the constraints that are needed for the game.

## [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)

Game, Team and Player classes inherit attributes from the Entity class. Entity class is the parent class. With this, it will confirm that a Team has Players and not the other way around. The same goes for the other classes, a Game has a Team, and GameService has a Game.The ProgramDriver class includes Main method and is directly associated with the SingletonTester. The SingletonTester confirms whether or not an instance exists with GameService. The SingletonTester and ProgramDriver have a relationship where the ProgramDriver uses SingletonTester when any executions are made of the application. These classes allow the game to play with multiple teams and 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)

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** | Good with user interface. Has better accessibility but you must have a Mac for development. Mac OS X server can be used on MAC is also affordable to execute. In my opinion however, Mac is not commonly used for these types of projects compared to Linux or Windows. Max OS X server located at [www.apple.com](http://www.apple.com) . Most programming languages come with the Mac OS. Terminal is flexible, which is an advantage as it can configure the server and can make changes. The implantation is diverse which has less restraints. A disadvantage is the that the cost is high with licensing and scaling. | Navigation of the platform is not comparable to other platforms as it has more user difficulties. Cheaper in cost and accessibility is simpler than other platforms. The Linux server is cheaper and is also an open-source server, which means there are more resources. Since Linux is not commonly used, experience with Linux will be vital in order to run on Linux server. Its terminal is flexible and is more cost effective. The license costs are low and the scalability is low too due to it being open source. Its technical restraints are low. | More expensive in cost, but user friendly. Server is more secure. Windows has Windows server. On Microsoft’s website, [www.microsoft.com](http://www.microsoft.com), it would be more in cost to execute, but it would be ready to function. Since Windows operating system is used more than any other operating system, there are those with the experience needed to operate the server very easily. An advantage is it high amount of available software, which means its licensing costs are low compared to Mac. | Much more convenient. Can be used as servers for development but could be very challenging to ensure development can be used on all mobile platforms. Affordable. Cannot host completing on the server due to lack of power, running a server on a mobile device isn’t the smartest, however it is the cheapest. It would be more cost-efficient running servers on mobile devices because one wouldn’t need much cost at all to get it started. It has a high scalability due to webhosting but needs a device that supports DNS. Its licensing is not expensive. |
| **Client Side** | Operating systems are not open source and depending on experience would determine how long it would take to setup. The cost would be very similar to Windows’ setup. The most common IDE for development is XCode for desktop applications. Web applications run off the browser and the other runs on a server. It supports today’s more modern web browsers but needs higher expertise. Has secure authentication that prevents risks of malicious hijacking, but it fails to support a large number of files like media. | Open source so cost of Linux should be very low, however it is recommended to have proficient experience with Linux since it is not used as much as Windows or Mac. This still has potential of being hard even with those who have experience. Operating system is free to the public. Supports minimal amount of file structures that can be accessible to the client without programming. Linux runs a fair amount of web servers, more than most would expect at roughly 70%. | Cost would be more expensive since this is not an open source. Cost would be similar to the Windows set up, but those who have proficient experience would need less time compared to those who do not have experience. Cost is more than Linux but is cheaper than Mac. Windows has a higher threat to technical issues and constraints because of the Microsoft’s web framework. | Affordable but takes more time to run applications created or other operating systems and platforms. Cost isn’t a factor as it takes little to no cost for set up. Experience may be in abundance, but time would be vital. More time will be needed compared to Windows, Linux and Mac because it would require multiple operating platforms and more than one mobile device. Has the ability to allow users to see updates at any point but tend to be harder to execute on other devices. In order to do this, frameworks have been created specifically to work without errors on a most mobile smartphones, tablets etc. |
| **Development Tools** | Mac can support most languages like HTML, CSS, JavaScript, Python, Ruby and also supports the libraries too for frontend. Since Mac computers come with most language interpreters. IDE’s such as Eclipse, Visual Studio, XCode, GitHub, NotePad++ but the most commonly used IDE is XCode with Swift being the language used the most. XCode is free. | There are many languages like HTML, CSS, and Java. The libraries support frontend and languages. The development tools compatible with Linux are GitHub, Eclipse, Atom, Visual Studio and NotePad++. The most commonly used language is Java and the Eclipse is the most commonly used IDE. Open JDK Java has no cost with personal development, but is not always up to date. | There are many languages are Python, HTML, Java, C++, C among many more. The libraries support frontend and languages. The most commonly used languages thought would be C and C++. Visual Studio is one the most commonly used IDE’s. Licensing depends on use. License cost is free with open-source platforms. Visual Studio is free to download. It is difficult to develop network-oriented apps with visual studio, but it is possible. | For iPhones, development tools are comparable to Mac. Most iOS apps use Swift but will need at least iOS 7 in order to be used for development. Android uses Java more as the language which isn’t supported with iOS. Both can run on all 3 machines, and the languages are HTML, JavaScript, CSS, while the supporting libraries support frontend. Other languages are java and python. Licensing costs are free with open-source platforms. Swift is open source. |

## 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**: After completing my research for this project, it is my professional opinion that Windows is the best operating platform to play Draw It or Lose It. Since it is the most used operating platform today, it would accommodate the needs of the users for efficiently compared to Mac, Linux, and mobile platforms. It has users who have proficient knowledge of Windows than the others. It provides more support with tools that the platform provides. Roughly 70% of users use then Windows operating platform, which will allow for more uses for the game. Although Windows requires manual modifications of files, and is not as flexible with hosting like Linux, Window is still strong enough to support a large number of users while also providing security and storage. Overall, Windows is just much simpler operating platform, and Windows’ server is also reliable with giving users faster and more reliable performance.
2. **Operating Systems Architectures**: Windows operating platform has a simpler GUI Design which will benefit Draw It or Lose It with the blueprint of the game’s design and can also be easily modified if needed. Windows also uses the NT Server which allows sustainability with gaming applications. It supports plenty of processors and also supports users who are remote. It lets users to connect without any restrictions. Windows server operating system architectures has many methods for the management of files and memory, which allows the user to assign the memory in a more ideal manner.
3. **Storage Management**: The storage management of Windows is simple and constantly updates the user with files that is recommended to be removed to allow for ample space. This lets the operating platform to run without the risk of losing storage space. I would also recommend Microsoft SQL with Windows because the game, Draw It or Lose It needs enough storage for online data so that a user will not need to download to their computer, they can play the game online instead. Microsoft SQL lets the data be stored online. This will greatly benefit the users experience with the game without downloading the game entirely to the user’s computer or to an external hard drive.
4. **Memory Management**: With game development, it is important to have a dataspace to store images which also ensures the ability to save them on another location of the computer. Not only is this good with images, but also with other files during the game development stage. Windows has a built-in memory management that will remove certain tasks or files to be removed to free up RAM for other tasks. With the game, Draw It or Lose It, since it is a web-based game, memory management is very important for the users. Since there will be plenty of storage and the management that comes with it, the speed of the game should not become an issue. Windows has automatic memory management that controlled by the system. Catching data also helps with memory management, which improves the overall playing of the game because it increases speed, performance, and accessibility.
5. **Distributed Systems and Networks**: A distributed system that uses a network is great when implementing new software and programs for a specific network. This communication between machines using the same network. The main feature of this is a simple line of fast communication between different machines on the same network. For Draw it or Lose it, it is recommended to have a strong and reliable network in order for the game to allow many users to connect to one server. This is what allows users to play in group player mode.
6. **Security**: Windows security is built in. This allows for software to be protected with scans to check for any threats and viruses. Having multiple layers of security will aid in the prevention of data loss or security breaches. Since Windows has multiple layers to the operating system, they push back potential malicious activity and threats. Windows has protection built in which detects and eliminates malware and viruses.