

Draw It or Lose it

# **CS 230 Project Software Design Template**

Version 2.0

## 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)4

[**Recommendations**](#_m8aleynsvzvc)6

## [Document Revision History](#_grjogdjh5fi8)

| Version | Date | Author | Comments |
| --- | --- | --- | --- |
| 1.0 | 3/19/22 | Chris Abbott | Designing the initial development of the product |
| 2.0 | 3/26/22 | Chris Abbott | Evaluating the different operating platforms from the client and server side along with the developmental tools used to operate them. |
| 3.0 | 4/11/22 | Chris Abbott | Recommendations made to The Gaming Room about different aspects. |

## [Executive Summary](#_sbfa50wo7nsh)

The Gaming Room would like to expand their application, Draw It or Lose It, to a web-based game that is able to run on multiple platforms. The team at Gaming Room is not familiar with these applications and have reached out to us for help. The game will have multiple teams that consist of multiple players. There may not be duplicate team names.

## [Design Constraints](#_2et92p0)

Must be able to alert the user if a team name already exists and to choose another one.

Games must be identified and categorized so that only one game can exist at any given time.

The system needs to be able to work with three different platforms: Android, IOS and the Web.

Each of the three platforms will be set up in a different way.

The system needs to allow different users from different platforms in the same game working together.

## [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 main method is in the ProgramDriver class. Entity class is the parent class of Game, Team, and Player. An object-oriented programming principle that is being used here is inheritance. Game, Team, and Player all inherit Entity’s attributes.

**"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)

| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| --- | --- | --- | --- | --- |
| **Server Side** | Mac is known for the client side but has a reputable server-side. It’s a Unix-based operating system. The benefit of using Mac is it has an easy interface. A weakness would be Mac servers will be expensive to maintain as well. | Linux is the most popular and is free to set up licensing. The Linux servers are very secure because they are customizable so companies can choose the security options that fit them. The weakness to Linux is that it is complicated and takes longer to learn how to operate it. | Windows is the easiest to set up and use but will be the most expensive server as well. This software requires payment for licensing. An advantage to having a Windows server is that it supports many applications. | Mobile devices would not be the ideal platform to use on the server side. Oracle is one of the companies that provide server roles on the mobile side. The weakness would be that mobile devices are not known for the server side so there is not a suitable network set up to back this operating system. |
| **Client Side** | Macs are easy to set up and work with, but it helps if you have Macs for everything and work with them together. Crossing platforms with Mac becomes difficult. The drives up the cost. Otherwise, the cost for the client could be generally low. | The cost for developing this would be high. You would want someone with an expertise in coding Python. From the client side, Linux is an open source which helps lower the costs. Windows and Apple are the most used operating systems though. | Windows would be very costly to set up everything. The support team would be set up to handle the mass volume needed. Windows is not an open source, but it is widely used by the population. | For this platform you need to be able to develop things in a smaller scale from a larger version and be able to break down what is not needed. This could be costly at times.  Everyone uses mobile devices so they will already be familiar with operating them it would just depend on which operating system they are using. Costs would be low for mobile. |
| **Development Tools** | FlexiHub would be an ideal development tool to work with when using a Mac. iOS is another tool used in operating a Mac. XCode, Atom, and Visual Studio are other common tools used by developers for Mac. | Python would be the programming language that is used to build this type of software for deploying on Linux. Seamonkey is an easy tool to use, and it has great compatibility with Mozilla products. Eclipse is a free and open-sourced IDE that is used with Linux. | Visual Studio is the program language used to build this type of software for deploying on Windows. HTML is a language widely used in web applications for Windows. | For mobile devices you would have to use each different program language that the operating platform the phone you are using operates. It could be FlexiHub or Python. They could also use other languages like C++. You could be using iOS as well. |

## Recommendations

1. **Operating Platform**: The operating platform that I would recommend using would be Windows OS. Windows is the most used platform out of the major ones, and it would be easier to use and get customer service help if needed. Since the application is web-based it would be better suited to handle things that come up on the back end such as troubleshooting. There are other operating platforms that are more cost effective, but I think having the freedom to do more is worth the money spent.
2. **Operating Systems Architectures**: Windows OS uses layers of architecture design and systems produced by Windows. This layered design system consists of two main components, user mode and kernel mode. The kernel is a bridge between applications and the data processing done at the hardware level. The kernel of an operating system implements the core functionality that everything else in the operating system depends on.
3. **Storage Management**: The recommended storage management system would be utilizing cloud storage. Operating a windows cloud server can be expensive but it allows for the flexibility of expanding the storage if needed for the web-based application down the road. This will allow us to increase the storage space without upgrading the hardware.
4. **Memory Management**: Memory management in this operating system is used to control the transfer of memory from the main storage to the primary disk during execution. With Windows OS, there are two options for Windows, physical and virtual memory. Virtual memory is more ideal because it will be able to store large projects without having to shrink them down to a certain size. Virtual memory is better protected than physical memory that is simply stored in the RAM of your processor.
5. **Distributed Systems and Networks**: Since the system is in the cloud if there is a connectivity issue or power outage the information and data would simply just move over to another server. This will allow you to host everything you can aside from the information on the user side. This will make sure that the running application will not have to stop if there were minor issues.
6. **Security**: iOS System Security is a very secure system. This system integrates the hardware and software with each other so that every component is secure. There are many levels to get through when trying to obtain information. This will be great to make sure the user can’t access any other information. Windows Defender antivirus program is a great way to protect information on the user’s computer. In most cloud-based services, like Google cloud, they will provide their own security to protect the information stored on their servers.