

Draw It or Lose It.

# **CS 230 Project Software Design Template**

Version 1.2

Brian Bertrand

Operating Platforms Project 3

CS-230

Dr. Vivian Lyon

## 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 |
| --- | --- | --- | --- |
| 1.2 | 04/15/2021 | Brian Bertrand | Phase three presentation. |

## [Executive Summary](#_sbfa50wo7nsh)

The Gaming Room has requested that Creative Technology Solutions assist in the development of a web- based version of their game Draw It or Lose It. Currently the game is only available on Android devices. Creative Technology Solutions would be required to develop a game that would be available on multiple platforms. These platforms would consist of development for use on Mac OS, Windows, Linux and Mobile Devices. The Gaming Room has requested the following technical requirements be met in this development process. Draw It or Lose It requires the ability to have one or more teams involved with each team having multiple players assigned to it. The game and team names must be unique to allow users to check whether a name is in use when choosing a team name. A final requirement is that only one instance of the game can exist in memory at any given time. This can be accomplished by creating unique identifiers for each instance of a game, team or player.

## [Design Constraints](#_2et92p0)

The Gaming Room would like to reach a much larger customer base. To do this their game must be developed to work on more than just the Android platform. Development of their platform would require different software tools and processes to complete as each platform has unique attributes. Testing of each platform would be required to verify that everything works as The Gaming Room requires. Development costs could be included as another design constraint as each process may differ for each platform. The time this process needs to be completed in can also have an impact on the development of the final product for The Gaming Room. Also, a key design constraint that is critical to this system is a security/login/identification mechanism to allow the application to identify and contain separate profiles for each person playing while also keeping their information secure.

## [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 UML diagram provides us with a way to map our course through the construction of the program for the game Draw It or Lose It by The Gaming Room. Through this diagram we can see all the classes and how they are connected.

Starting at the top of the diagram we see the ProgramDriver class. This is the application start up program containing the main method. The ProgramDriver and SingletonTester classes have an association relationship. ProgramDriver can reach Singleton tester and this enables the main class to complete testing.

The four other classes shown below the Entity class (GameService, Game, Team and Player) are all associated with each other. The 0…\* shows that the classes can be associated with zero or any number of each other. The entity class is shown to use Aggregation from the Game, Team, and Player classes indicated by the hollow arrow pointing toward it. This means it is an object that has a game, team and player. The GameService class alone inherits from the Game class.

****

## [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** | The Mac server system presents itself as a user friendly and efficient system. It has many characteristics of the Linux server system but there are limits on what you are allowed to access and adjust. A disadvantage to this would be cost. Compared to other cost-free options, this could be something The Gaming Room may not want to cost them over time for Apple dependent support and running cost.  **Highlights:**  +Unix OS  +Open Source Servers  +Secure  +User and File access controls  +Support LDAP and ADP  -Docker support is virtual only  -hard to apply in server only setup. | Linux for servers provides a cost free and very secure system. You are not limited in the ways that you chose to adjust and use the system. A disadvantage could be the steep learning curve when starting out. This system is a very effective and popular choice providing you have skilled people operating and maintaining the system.  **Highlights:**  +Unix OS  +Open Source Servers  +Secure  +User and file access controls  +Support LDAP and ADP  +Docker Support  +Large online/cloud support.  -Need to decide on Redhat or Ubuntu support. | Windows server systems are a popular choice with a large support system. There are costs for operation and support on this system, but it is usually less than systems offered by Apple. A disadvantage to a Windows server system would be vulnerability. This would require more support and additional software to protect this system.  **Highlights:**  +Secure  +User and file access controls.  +Support ADP  -Need to use server version.  -Limited LDAP  -Poor open source server support. | Hardware limitations would greatly reduce the capability of using a Mobile Device for the server side of a game. A more stationary and accessible platform would be better for their system and would provide better end user performance and stability.  **Highlights:**  +iOS developer tools.  +iOS multi-device support  +Android developer tools.  -Android poor multi-device support. |
| **Client Side** | Developing the game software for the clients that would be operating the game from Mac hardware would be the Highest in cost compared to other alternatives. This would also require developers with a more specific skill set as this is a less popular platform for game users.  **Highlights:**  +Support Safari browser.  +Firefox  +Chrome  -No Internet Explorer.  -Small install of client desktops. | This would provide the lowest cost option for development. The skillset required would be high and the demand for game playing on Linux has been low but over time seems to be growing. There would be less support for the development of the game for Linux as the developers of the platform did not engineer it for this purpose.  **Highlights:**  +Support Firefox  +Chrome  -No Internet Explorer  -Small install of client desktops. | To develop the game software on Windows would be lower in cost than Mac. There is a tremendous amount of support for gaming and development on Windows systems. This would require less expertise than all of the options available due to so much focus on this platform having already been completed.  **Highlights:**  +Support large installed base of client desktops.  +Firefox  +Chrome  +Internet Explorer  -No Safari | With the popularity of Mobile Devices and the accessibility the users have to operating the mobile games, this platform would be fast to deploy and test. For each mobile operating system (iOS and android) there would require a specific skill set and development process. The Gaming Room has already developed the Android game so this would require the iOS app to be completed.  **Highlights:**  +iOS: Firefox, Chrome and Safari.  +Android: Firefox and Chrome. |
| **Development Tools** | Mac is able to support many of the same development tools as the other system options. Some of these tools are required when developing the application to work on a Mac. Swift is a language unique to Mac applications that is developed and maintained by Apple using the XCode environment. This may require specialists familiar with XCode and Swift. Some of the development tools may need to be purchased and others are free for this system. Mac can also support the use of many other widely used languages such as HTML, Javascript, Java, Python, C++ and many more.  **Highlights:**  + iOS native and Android development supported + High-productivity environment + Large language support including Swift and objective-C + Java support + Unix shell scripting  - Small pool of developers  - Difficult /impossible to test IE browsers - Few IDEs, but high quality | Linux has a very good variety of cost-free development tools available. The developers for this system would need to specialize in Linux/Unix development and this has the steepest learning curve of the development required. Eclipse is a very popular choice as an open source free development environment. Linux will also support the use of HTML, Javascript, Java, Python, C++ and many more. With add on programs you can develop Mac apps on Linux, but you would need a Mac system to run them.  **Highlights:**  + Large developer pool  + New technology first platform  + Largest set of IDEs with quality but no always support  + Large language support including Swift and objective-C  + .net support  + Java support  + Unix shell scripting  + Powershell  - Android dev support only: native | Due to its popularity there are more development tools and developers available for Windows than the other operating systems. It is easier to operate than Linux with lots of support. Windows development tools will support many of the same Development environments including Eclipse, but they offer Visual Studio that works best on Windows. Visual Studio is a powerful system, but it does include licensing costs. Languages supported match other operating systems including HTML, Javascript, Java, Python, C++ and more.  **Highlights:**  + Large set of IDEs with support and quality  + Large language support including C#  + .net support  + Powershell  + Java support  - Android dev support only: native | For Development tools on mobile devices, some of the same ones will work that are used for the other platforms. Eclipse is a very popular and capable choice that is free. Swift is the most popular choice when it comes to creating iOS apps. It is preferred for its powerful tools that enhance app safety and system performance. To use XCode on Apple systems to develop there is no cost for the software but to acquire the Apple hardware for development carries a higher cost than other systems. A more focused skill set would be required to develop for mobile application development. |

## Recommendations

1. **Operating Platform**:

To provide The Gaming Room with the server operating platform that will provide the most features I would recommend a Windows based system. More specifically the Microsoft Azure cloud server system has proven to be highly popular, cost efficient and very well supported. One of the many features of Azure is its scalable cloud storage that can be adjusted if The Gaming Room requires more as the game platform grows when hosting the game for more users and their variety of devices.

1. **Operating Systems Architectures**:

Azure is an established system that The Gaming Room can rely on. It is also available at different price points depending on the scale of the deployment. The Gaming Room will not need to worry about paying for features they do not require. As they state on the Microsoft Azure website, “Azure App Service is a fully managed web hosting service for building web apps, mobile back ends, and RESTful APIs. From small websites to globally scaled web applications, we have the pricing and performance options and that fit your needs, including new Reserved Instances on Premiumv3, which offers savings up to 55% versus pay as you go.” (Microsoft Azure, 2021). They provide the means to help your system grow and want to grow with you as you succeed.

1. **Storage Management**:

The recommended Azure system provides scalability that is available right away as The Gaming Room customers require more for increased numbers of players on a variety of systems. This would reduce downtime or performance concerns to the existing customers currently operating the game. The ability to adjust and monitor this would be exceptional using the tools provided by Azure. If the user prefers, they could also use local disk, NAS or cloud storage for their personal storage of data.

1. **Memory Management**:

With Azure’s scalability, the system will always provide what is required for your deployment and it can rapidly adapt as more resources are needed. This dynamic allocation will help maintain performance and reliability required for The Gaming Room. The memory required needs to be sufficient enough to prevent paging.

1. **Distributed Systems and Networks**:

For the game “Draw It or Lose It” to be deployed, there are some details to the design of the distributed systems and network to consider. The game will be distributed to the required platforms and would be designed to operate on Mac, Linux, Windows and Mobile Devices on the iOS and Android operating systems. Each of those operating systems would require a development process to adapt the game to be compatible with that system.

Once this is completed the server deployment will be designed using a REST API configuration. This design will allow all the different client operating systems to interact with the server hosted system using a single language and single Microsoft Azure host cloud server. This system simplifies the server side of the game and will create a more stable environment that is easier to maintain and expand. Microsoft’s Azure server system will offer a strong support system that will rapidly provide alerts, recommendations and repairs to any problems that may arise due to performance concerns, power outages and network connectivity issues.

1. **Security**:

“The Microsoft Azure Security Center provides many built-in capabilities that are organized in six functional areas: Operations, Applications, Storage, Networking, Compute, and Identity.” (Microsoft Azure, 2021). The Security Center is a very supportive and interactive system that provides the customers and developers with great visibility and can quickly detect and prevent threats to The Gaming Room’s system. Another feature to Azure is encryption in transit and at rest for all data that is transmitted between the client and server along with any data that is stored on the Azure system. We highly recommend Azure and believe it will provide the benefits we have mentioned and many more to the developers and players of the game “Draw It or Lose It”. With the support of the Azure system the customer should be provided with vendor security updates along with vendor -specific security software. They should also provide the widest support of encryption protocols and certificate authentications and the Widest support of SSL protocols for client/server with hardware acceleration.

# References

(2021, 4 17). Retrieved from Microsoft Azure: https://azure.microsoft.com/en-us/services/app-service/#security