



Draw It or Lose It  
CS 230 Project Software Design Template  
Version 1.0

## Table of Contents

CS 230 Project Software Design Template.....	1
Table of Contents.....	2
Document Revision History.....	2
Executive Summary.....	3
Design Constraints.....	3
System Architecture View.....	3
Domain Model.....	3
Evaluation.....	4
Recommendations.....	6

## Document Revision History

Version	Date	Author	Comments
1.0	05/23/2021	Wesley Page	Information for the executive summary, design constraints and the domain model.

## 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**

“Draw It or Lose It” is a game currently only available on the android platform. The Gaming Room wishes to develop a web-based version which can run on multiple platforms. The game consists of one or more teams playing 4 rounds, each round being a minute and players guessing an image pulled from a preloaded library as the image renders over 30 seconds; if the playing team fails to solve the puzzle, then the opposing teams have a chance solve rendering puzzle over an additional 15 seconds.

## **Design Constraints**

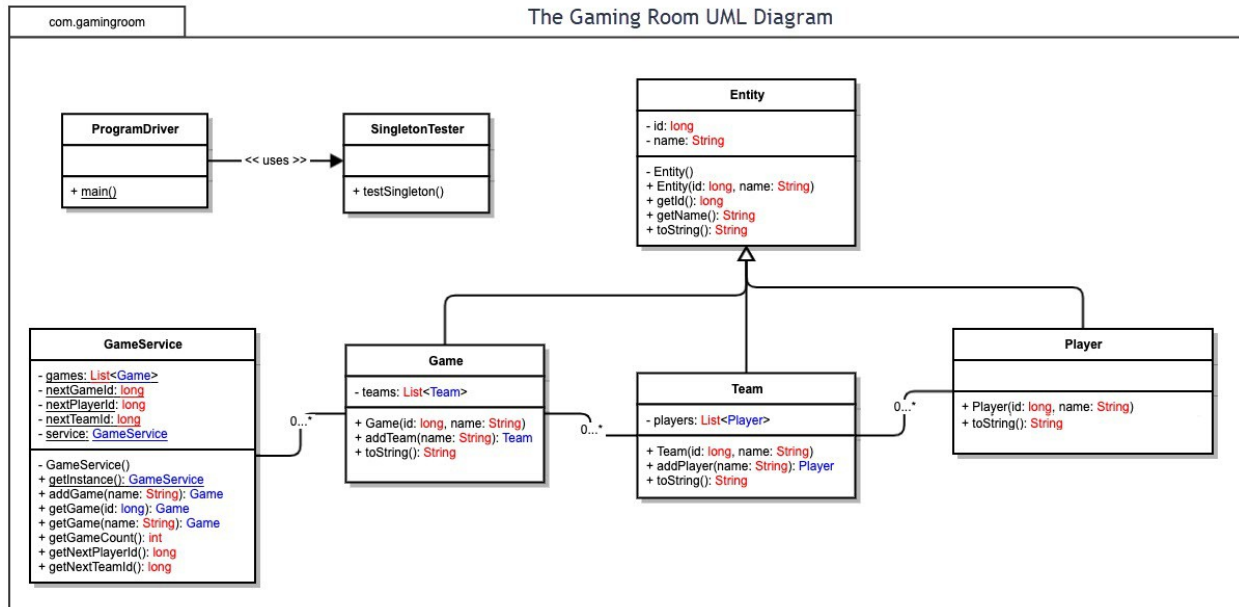
Each team must have multiplayer capability, with a single unique instance of the game existing in memory at any given time with a unique instance of the game, team, and player. Team and game names must be unique to allow players to choose a team name. Android, IOS, and the web each have differing development kits and the API will need to be tailored to work on each platform. Staying within both the timeframe and budget for the project will be critical to allow for feedback to be implemented from the client as well as end-users.

## **System Architecture View**

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**

The ProgramDriver class contains the main method and is where application execution takes place. ProgramDriver uses directed association with SingletonTest to test if there is already an instance of GameService, so GameService can only have a single unique instance at any given time. Entity class is the parent class to the Game, Team, and Player classes, making it a super class, where each child class of Entity inherits this super class's public attributes. GameService references the Game class, Game class in turn references Team class, which references Player class, where each of these references are made via aggregation. A Player may not have a team, but a team can have a Player. A team cannot have a Game, but a Game can a Team. A Game cannot have a GameService, but a GameService can have a Game. Each Team may not have any duplicate players, only a unique player at any given time.



## Evaluation

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.

<b>Development Requirements</b>	<b>Mac</b>	<b>Linux</b>	<b>Windows</b>	<b>Mobile Devices</b>
<b>Server Side</b>	MacOS Server is a possible choice and offers similar functionality to Linux as well as better compatibility with Mac devices and terminal commands to configure the server. However, it would require someone operating it that has experience, while this also comes with the benefit of getting the most out of the server it would also drive up costs, in addition there are licensing fees.	Linux is a popular OS for running a server, and it is similar to a MacOS wherein a command shell is used, unlike a MacOS it doesn't have licensing fees. Given it's wide popularity as a server finding experienced administrators make this a cheaper and high quality option.	Windows is a strong option. It has the most accessibility of the Server options and is the most well-known OS and already has many built-in applications that make administration of the server relatively easy to get high performance. This is the most expensive of the options, as licensing fees and initial setup can be significant.	Mobile Devices don't have the processing power necessary for hosting a server.
<b>Client Side</b>	Moderation amount of expertise and time required. Cost is similar to Windows. Familiarity with the OS would be needed to ensure coding was compatible. This platform would still be required for any application to run on any Mac device.	Although it is open source and free, it's still not widespread, so given that it's a less widespread platform an expert would need to be on hand to ensure compatibility.	One of the theoretically time friendly options. Cost is similar to Mac. Given that Windows is a widely used OS, the development team should be familiar with it. There are many tools available to help support development.	Mobile devices are widely worked with and shouldn't negatively impact development time and sourcing expertise for developers working in Mobile App development shouldn't be as much of an issue.
<b>Development Tools</b>	IDEs for iOS include Eclipse, specific versions of Visual Studio, iOS SDK, Xcode. Languages include swift and objective C, but you can run any language for which an interpreter exists (like Java)	IDEs and development tools include Eclipse, NetBeans, Atom, Vim and Gedit, among others. Most programming languages are supported. C, Python, Java, Ruby, Perl, Rust have cross-platform reference implementation with Linux.	IDEs and development tools include Visual Studio, Eclipse, NetBeans, Xcode. Languages are widespread. Similar to Mac, any language could be used, as long as it has compilers & interpreters that support Windows targets.	Tools exist allowing for development for Android with Android Studio IDE and SDK, and associated VMs to test apps. Similar proprietary tools exist for development in iOS environment for iPhones.

## **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:** For the Gaming Room's "Draw It or Lose It" application, the Windows server operating system is recommended for the best results and expansion of the application. Using Windows server provides users with ease of use and ensures reliability with a protected and supervisor mode. The Windows server OS also supports many server roles, including web server, file server, application server, mail server, database server, printer server and more. Windows operating platform also offers software engineering abilities for developers.
2. **Operating Systems Architectures:** Windows server OS architectures include a variety of concepts for memory and file management that allow users to control and coordinate memory to best suit specific needs such as allocating memory to different programs and/or free space when it is no longer needed for later use, and it offers ease of use with a simple GUI theme. Windows server OS can utilize multi-processor scheduling to optimize performance on machines with multiple processors. Windows has many software packages available, offering a wide variety of programming options for developers. Command Prompt power shell utilization offers quick and easy server configuration settings for regular maintenance across the company's computers/machines. Windows OS also provides developers the options to work with an array of different programming languages.
3. **Storage Management:** Windows server operating system offers simple configuration settings and allows for convenient memory management and allows for routine relocation of chunks of information to solid state storage to open up more storage for user needs, this ensures there is ample storage space. Users can also read/write company files on both personal and work devices though copying to servers in the data center.
4. **Memory Management:** Memory management options offered with Windows server OS includes physical and virtual address space allowing up to four gigabytes of memory as well as means for running applications smoothly. Supported pagefile enables the system to move pages of virtual address spaces to the hard drive of the system which allows the random access memory frame to free space for additional uses.
5. **Distributed Systems and Networks:** Distributed system and networks can have common issues like queuing repercussions as well as routing and congestion problems, however, these systems offer easy communication and coordination between various processors and individual workstations. Some common problems in relations to using distributed systems include independent failed components, absence of global clock, simultaneous computation of components, which can result in lagging computing performance and connectivity problems among individual users. Nonetheless, a useful feature associated with distributed systems and networks includes the user ability to communicate with different servers (i.e. web servers, data servers, et cetera). Computers on the same network can run more efficiently as tasks can be divided among the users and the server. A distributed system with load balancing can defend against connectivity issues with redundancy, and Windows server OS includes an Azure-

inspired load balancer. If an app server instance is unavailable, the load balancer can direct traffic to a good server and prevent downtime. A distributed database solution can continue to serve the application when other instances are down. And promotes continual deployment with zero downtime. In this instance new code could be deployed to application servers in sequential fashion while leaving other application servers up and running to serve users.

6. **Security:** Windows operating platforms provide clients with user account control settings that help to secure data going into and out of the system. It also ensures that authorized operating platform modifications are not made without acceptance from administrative user(s). Windows offers shielded virtual machines where unauthorized access to protected data is prevented by the host administrator(s), and it has the Windows Defender Application Control which ensures the control of which applications can run on the machine. Windows server OS has built-in protection against memory corruption attacks and Windows defender which includes a firewall and malware solutions for known malware threats. It possesses VPN service capabilities which help to protect the client information and history from being used for malicious purposes; RDP stream cipher that calls encryption protocols for smaller amounts of information: i.e. passwords, credit card numbers, social security numbers, et cetera. Alternatively, Windows OS offers Microsoft DirectAccess for remote positions and work sites. This feature makes use of authentication and auto-encrypted ESP when users connect to business networks. The Advanced threat analytics feature uses Active Directory network traffic as well as SIEM data to locate and notify the user of potential threats.