

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)3

[**Recommendations**](#_m8aleynsvzvc)5

## [Document Revision History](#_grjogdjh5fi8)

| Version | Date | Author | Comments |
| --- | --- | --- | --- |
| 1.0 | 01/22/22 | Gage Warrington | Create a web-based game |

## [Executive Summary](#_sbfa50wo7nsh)

The client Creative Technology Solutions (CTS) has hired a new client Gaming Room who wants to create a new web-based game that can be used on multiple platforms. The game is titled Draw It or Lose It and is currently only available in an Android app. The purpose of this game is the application will render images from a library of stock drawings. Teams will take turns each round to guess the puzzle. If a puzzle is not guessed correctly the remaining teams have an option to offer one guess for each puzzle.

## [Design Constraints](#_2et92p0)

* Must have more than one team involved
* Must have multiple players assigned to one team
* Game and team names must be unique so users can check whether a name has already been chosen
* Only one instance of the game can exist in memory at any given time.

These requirements have been given to us by the client and must be followed when writing the code. Since we are already on a mobile app for Android, we need to create it for other mobile systems. As well as computer systems like Windows, Apple, and Linux. This can be done by either changing the code for various operating systems or attempt to create the code so it can run on all systems.

## [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 Entity class creates a relationship between Player, Team, and Game classes. They will get information from the Entity class. We can see that each class is going to have Id and name since they are connected to Entity, this would make Entity a super class. We can also see that Game service is connected to Game which is connected to Team which is connected to Player.

**"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 must 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 commands to configure the server. Very upgradable. Does not work well for web hosting. Known for flexible terminal. | It is open sourced. Usually free or very cost friendly.  Security is better as it gets caught early. Not as secure as Windows. Not so easy to find supporting applications. | Uses proprietary sourcing. very secure and has great security. Not free and can be very costly. More viability of software. Has much support with many techs support options. | Page loads quickly as there is no “white page”. More cost effective and more compatible. Poor security. Content visible to search engines. |
| **Client Side** | Less user friendly than Windows.  More expensive than low-end windows devices.  Built in client security. Requires an average amount of time to understand support. | Way less user friendly than windows and Mac. Can be run for cheap. Eclipse IDE is most popular. C language is the most popular option. Need maximum time and proficiency to support Linux systems. | Very user friendly. Cost can be cheap or expensive. Built in client security. User does not need a lot of time to understand how to support Windows setup. Minimum expertise and time required. | Can easily be updated by client from anywhere they bring their phone.  Hard to perform applications from other platforms.  Slightly harder to implement than other devices. Can update the screen instantly when the user clicks. |
| **Development Tools** | The development tool used is XCode. The language used is Swift. Other languages can be used like HTML, CSS, and JavaScript, ETC. | Can work with Visual Studio and Eclipse. Uses a programming language called Visual Basic. Other languages can be used like HTML, CSS, and JavaScript, ETC. | I can run many IDEs as well as Visual studio and Eclipse.  Other languages can be used like HTML, CSS, and JavaScript, ETC. | Easily create apps using Android and Swift. Has development environment Android studios for Android. Apple has its own IDE called XCode. Other languages can be used like Java, Koltin, HTML, CSS, and JavaScript, ETC. |

## 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**: When we talk about operating platforms, we want to understand that some are designed to just fit the needs of the current system they are running. Good examples of this are consoles such as PS4 and Xbox. Their operating system is designed to fit the needs of those systems. They all have their own characteristics to better their system to compete to keep their platform continuity. Right now, I would recommend Windows as it reaches out more to other platforms than other systems do. Windows also is the owner of Xbox which has a service called game pass allowing Windows users and Xbox users to access free games on both platforms. Windows also has great server capabilities and great tech support since it is such a huge company.
2. **Operating Systems Architectures**: Windows has great applications that allow programs to show a Graphical User Interface. Windows also has great technical support as it is used by various systems daily. Windows has two options for its architecture. 32-bit and 64-bit, 32 is most likely to work with older hardware and software. There will be performance and features that will be missed with this option. 64 will work with today's hardware and software and comes with all the newest features.
3. **Storage Management**: Windows 10 comes with the feature storage sense. This is a built-in feature that allows you to manage the files on your hard drive but also allows you to see valuable information like how much space a file takes up. Windows also has cloud storage so you can utilize less hardware space and give yourself ease knowing your files are backed up too. Windows also has built in features to detect junk files that are taking up unwanted space. This can be done on a server and on a personal computer. On the personal side the storage will be up to the user. For fast reading and writing the best option is SSD (Solid State Drive) as it is the fastest option currently available. There is still the option for HDD (Hard Disk Drive) and that is a good option for a budget.
4. **Memory Management**: For this game we will have to create a library of lots of various pictures. To limit the amount of memory we will want to allocate the pictures outside of the default pictures folder and put it together with the project files. Windows also comes with a process that has its own address space that allows for addressing up to 4 Gigabytes of memory at a time. It allows all the threads of the processor to access the address space. Furthermore, it has a corruption blocking feature by not allowing threads to access memory that currently belongs to another process.
5. **Distributed Systems and Networks**: The company needs to find a program that will take the code they have for the game and alter it to fit various platforms. This can then be installed on a server so that all platforms can use the game. Once on the server the company should make sure that they can handle that amount of traffic. Network games allow players to join over the network regardless of the system they are using. They are multiuser systems that have a shared database that allows players to interact with one another over the network. A good program currently available to create network games is Artery. It supports network games by providing a program interface and optimizing network performance by using application semantics. The only problem is you must begin from nothing so the work is going to be hard but for online games it will make it worth it.
6. **Security**: Windows does come with a built-in security system. It would still be in the best interest of your company to get an external one as well. The pre-installed Windows security can scan for viruses and various kinds of malware. This will all happen in the background and can keep the user's information safe. There is also data controller software that manages the data coming in and out of a server keeping players' personal information more secure. Keeping passwords to online accounts to access the game is important as well as being the leading line of defense in allowing unwanted people in. If playing an online game from your phone or personal computer, it can be damaging as once in, they can access your financial and personal information. Playing from a console is less risky as there is usually less personal information on a console.