

Draw It or Lose It

# **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 | 0921/2022 | Brandi Noell | Develop a web based game |

**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 client, The Gaming Room, has requested help to develop a web based game, that will run across multiple operating platforms. The game is to be called Draw It or Lose It. And this game is only available on the Android platform. The game is played with multiple teams. Each team has multiple people that play on the team and each round is four minutes long. The teams compete to guess what is being drawn.

## [Design Constraints](#_2et92p0)

* The game will need to run on multiple platforms.
* Each team needs to have multiple people
* Need to have multiple teams
* Only one version of the game can exist

This will be the requirements for the game. The company would like the game to run on multiple platforms and devices as well. A code will need to be written to and rewritten to perform on multiple platforms.

## [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 is the super class. This creates a relationship between the Game, Team, and Player class. In the UML, we can see that the Game Service refers to the Game Class, and the Game Class is connected to the Team Class and Player Class for references as well. The Singleton Tester and the Program Driver class is where the program will get it’s execution functions.

**"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 has a flexible terminal, is inexpensive to implement. But MAC OS is not as popular. But users are very loyal to it. | Linux is low cost  Linux is secure.  The downside to Linux is it is more difficult to find support for it for applications. | Windows offers the most users.  There is high number of resources for Windows.  Windows can get a lot of viruses. | More popular than a single platform place.  Can reach more users  Has poor security |
| **Client Side** | Will require a lot of expertise and time. The cost will be close to windows.  A person skilled in MAC wouldn’t need as much time to run it, versus someone little experience will need more time. | High Level of expertise. Will cost minimal.  Will require maximum effort and knowledge to use it. | The cost will be close to MAC.  A person skilled in Windows would not need as much time as someone not as skilled in Windows. | High Flexibility.  Lost cost. There will be more time needed, as it will be harder to implement on other devices. |
| **Development Tools** | The easiest used program would be Swift. Others like Java C++ and other IDEs require a bit more support. | Eclipse works great along with Atom on Linux. Eclipse supports Java. | Most common languages work on Windows. They are easier to implement and switch between different IDEs | Android can be used for developing multiple apps. Multiple computer languages will run on Android as well. Apple is less lenient. |

**Evaluation**

**Server Side:**

Does each of the operating platforms offer a server-based deployment method where the website will be hosted? Yes, there will be multiple software supports for this.

What are the potential licensing costs to the client, the Gaming Room, for the server operating system. If they go through Linux, that will be their cheapest option, but the language for Linux takes a lot of understanding, and the developer will have to have Linux skill base. If they go with a more user friendly development like Windows, there is more support, and familiarity even though it will cost more for the license.

**Client Side:**

Windows provides Visual Studios and Visual basic for programming language and a development tool for IDE. Windows is going to offer more support and be more secure for using as a development interface, but it’s going to be more expensive. Linux on the other hand is going to be open source and cost less, but it will be less secure and there will be less support.

**Development Tools:**

A web application can run on a web browser while effectively running on a mobile device as well, and the other one would run on the server. The developed program would have to run both in HTML and java though efficiently and effectively. Web apps are usually used on desktops, which offers more space to use on a screen the interface could be similar to the mobile version but offer more options and be spread out over a bigger interface.

## 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**: I would recommend using the operating system of Windows OS. Windows seems to be most widely used on computers and provides easy access for users. Windows would also allow for developers to format their own software for the game.
2. **Operating Systems Architectures**: Windows OS has a simple user interface and can enable the Graphical User Interface. There are many software packages available for Windows OS, that allow the software packages to be edited, developed on, and used in a variety of ways from a developer’s aspect. You can also use different computer engineering languages on Windows OS.
3. **Storage Management**: Windows allows you to choose your storage locations, create different folders to divide data and store in multiple locations. Windows OS also offers cloud storage, which provides developers with ample storage space. With Windows OS you can also compress files and determine how much space they will take up on your hard drive.
4. **Memory Management**: With the overall purpose of this game, memory management will need to have ample room and need to run smoothly for the hundreds of pictures the game will require to save. There will need to be a database for these pictures and using memory management will allow for easy storage.
5. **Distributed Systems and Networks**: Each operating system is a bit different, but the game can be designed to work and interact across different platforms. A couple problems that could arise when trying to do cross platforms is congestion, but these systems can be easily fixed to communicate with one another. Developer four can help run cross platform software. It is a cross platform software, or IDE that allows multiple platforms to talk to each other and run the program necessary to run smoothly on the current user’s operating interface.
6. **Security**: Windows OS offers plenty of security protection software. Some are built in, like firewalls, and others can be downloaded and installed. You can also put antivirus software on Windows OS to protect from outside threats. The operating platform can scan for virus, security threats, malware, and anything else that poses a security threat to the game and overall operating system. Customers can use VPNs which protect the client’s information from being used by malicious software.