

Draw It or Lose It

# **CS 230 Project Software Design Template**

Version 1.2

## 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 | 11/14/21 | Hillary Loyd | Initial Design Template to add in Summary, Constraints, Domain Model |
| 1.1 | 11/28/21 | Hillary Loyd | Inserted Evaluation |
| 1.2 | 12/12/21 | Hillary Loyd | Inserted Recommendations |

**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 Gaming Room wants to develop a web-based game that can run on multiple platforms. The game will be called “Draw It or Lose It” and is currently only available on android. The purpose of this game is multiple teams consisting of several people going four rounds at a minute each. When a picture is pulled from a library of images one team guesses till time runs out. If not answered each opposing team member gets to answer till 15 seconds runs out.

## [Design Constraints](#_2et92p0)

* Needs one or more teams involved
* Each team has multiple people
* Game and Team names must be unique to allow users to check whether the name is in use or free
* Only one instance of the game can exist at any time.
* Must run 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)

Entity creates a relationship between Game, Team, and Player class. This means they all inherit or get information from Entity. With UML we can show this with inheritance. So, each class will share common references like “name” and “id”. Making Entity a superclass. When we look at their relationship, we see Team and Player is a “has a” type. While Game has a Team and GameService has Games. When we use UML, we call it aggregation (HAS-A). When a user “has a” I mean it's an instance of one class and has a reference to an instance to another class. When we look at this diagram, we see GameService has a reference of Games, Games a reference of Tea, and Team a reference of 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)

| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| --- | --- | --- | --- | --- |
| **Server Side** | Flexible terminal commands to configure the server, access, or make changes.  Characteristics: It is popular in web hosting  Advantages: It is upgradeable, it has various options for different web hosting requirements  Disadvantages: It is less preferred for web hosting services | Flexible terminal commands to configure the server, access, or make changes and more cost friendly.  Characteristics: Secured, most preferred.  Advantages: Security flaws are caught before they become an issue, it is the most preferred choice for web hosting services  Disadvantages: It is more difficult to find applications to support the web hosting required needs. | More software available compared to other OS.  Characteristics: It is dominant to the other platforms. Close platform  Advantages: High resource requirements, less loading time, high comfortability  Disadvantages: Easy virus susceptibility, poor tech support | It's better if the server is immobile and can be tracked in a single place. Specifications are better in other devices.  Characteristics: More popular, high portability.  Advantages: Have a wider reach, better compatibility, cost-effective  Disadvantages: It is highly selective to various smart mobile devices Poor security |
| **Client Side** | Moderate expertise and time required.  Cost: Expensive.  Accurate skills needed to navigate OS. | Maximum expertise and time required.  Minimum cost.  Linux data is required to use the OS. | Minimum expertise and time required.  Cost comparable Mac.  Easy to navigate and understand how to support windows. | Minimum expertise and time required.  Cost comparable to Mac.  Provides flexibility to clients or even developers to see updates at any place. Slightly more difficult to implement than other devices. |
| **Development Tools** | While mixing in nice tools like notepad++, Macs can run all languages. | Linux can work with Visual Studio, Eclipse, along with notepad++ for a nice and easy-to-use tool, as well as many more tools. Languages can consist of C, C++, CSS, Java, JavaScript, HTML, PHP, Perl, Python, Ruby, or Vala, Linux supports them all. | Easier to use than Linux but can run the same as it. So Visual Studio, Eclipse, notepad++ to name a few of the many tool’s Windows can use. Languages can consist of HTML, CSS, JavaScript Java, Python, PHP, and Ruby and the main of C++. | You can create countless apps using android and swift. Java is the official language for Android App Development and consequently, it is the most used language as well |

## 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 The Gaming Room starts on windows devices as it has more software available along with minimum expertise and cost to get projects going. You also won’t run into a shortage of IDE’s to work with.
2. **Operating Systems Architectures**: Windows offers services used with the aid of all Windows-based applications that allow applications to exhibit a Graphical User Interface (GUI) whilst gaining access to system assets and much more. These applications additionally refer to Graphics and Multimedia, messaging, and net services. These services can be used by the usage of a consumer account or a server specifically.
3. **Storage Management**: Windows 10 comes with an exceptional function known as storage sense. This permits you to scrutinize and control documents on your hard drive, alongside with how much space it takes up. Other elements encompass being capable to select and save locations for apps making them less difficult to find. And just like different dives, you can additionally use the cloud to store data. The built-in storage system permits for convenient file creation and placement for massive projects, so they won’t get misplaced or carelessly deleted.
4. **Memory Management**: While creating this game you will need to create a database or library with a lot of pictures. Memory allocation permits for convenient storage of images outside of the default image folder. This permits your entire project to be stored collectively in an extra secure area on your computer.
5. **Distributed Systems and Networks** It’s an IDE that can be run on any device. Once the game is created you can simply export the game file into the web, iOS, Android, and many more selections that will permit cross-play. This will assist with dependencies. To prevent different issues like outages or connectivity, the organization will want to make certain their servers are sturdy enough to help massive player volumes alongside with backup power options for power outages.
6. **Security**: Windows comes with built-in security protection software. The pre-equipped protection can scan for malware (malicious software), viruses, and security threats. This all occurs in real-time, and since threats change, the system updates automatically to maintain the system and user information are safe. However, to secure user data and information it would be advocated to use another source.