

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 | 10/15/2021 | Darkenley Minviel | Evaluation of expansion on new operating systems and updated recommendations. |

## [Executive Summary](#_sbfa50wo7nsh)

*The Gaming Room(“The Client”) would like to expand their gaming application to multiple operating system using different software patterns in a distributed environment. Their game “Draw It or Lose It” is currently available on Android only. 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)

Development constraints:

*Security of user information*

*More than one team involved*

*Team contains multiple players*

*Game and Team names must be unique to allow user to check whether name is in use or free*

*Only one instance of the game can exist at any time.*

*Must run on multiple platforms*

*General constraint:*

*The Gaming Room would like this to run on all devices.*

*Other mobile platforms than android .*

*Windows, Linux, and IOS versions*

## [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 classes Game, Team, and Player inherit from the superclass Entity which allows the attributes and methods in the Entity to be used by these classes. Game, Team, and Player are associated with each other using a zero to a many associations which allows them to use many instances of the others. Finally, the ProgramDriver class is the driver of the program, and it uses the SingletonTester while being executed.

**"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 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** | GUI friendly to developers. Mac has flexible workstation for easy accessibility and server config. | Difficult platform. Linux uses command shell for simple server configuration and accessibility. | User friendly platform. Windows uses command. And there is an infinite amount of software available for windows. | Mobile device spec. vary: Android , IOS. It will be challenging to develop a game that compatible with all mobile platforms. |
| **Client Side** | Time necessary for Mac support is average. Appropriate skills will be needed.  The cost of Mac is more expensive than the other OS | Time necessary for Linux support is max.  Deep knowledge of Linux is required. The cost is minimal. It is more cost effective compared to Mac. | Time necessary for Windows support is minimal. Minimal knowledge of Windows OS is required. Cost is like the cost of Mac. | Time necessary for Mobile devices support is max. Deep knowledge is required. Cost is maximal. |
| **Development Tools** | JavaScript, CSS, Python, HTML, etc…  Tools include libraries to support libraries and the IDES. | Ruby on rails, Java, Python, CSS, JavaScript, HTML, etc…  Tools include nodejs, Github,  Command prompt, Repl.it, and Visual Studio. | Ruby on Rails, Java, Python, C++, C#, JavaScript, HTML, etc…  Tools include Eclipse, command prompt, Visual Studio and Repl.it | Swift, Android, Java, Python, CSS, JavaScript, HTML, Ruby on Rails, PhP, C++, etc…  Tools include Repl.it, Visual Studio, Command prompt, nodejs, Github. |

## 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**: Based on research, Windows is the recommended Operating platform. It provides users accessibility to many software packages and the ease of use, and their cost is typically low. It is considered to be the most used operating platform which means the most typical users have more knowledge of windows.
2. **Operating Systems Architectures**: Windows OS needs an activation code. It provides the ease of use through an effective GUI design. It also provides a lot of options to programmers through many of the software packages. CMD can be used for server configuration. Its architecture allows applications to use the platform’s kernel processes without affecting those processes.
3. **Storage Management**: Windows OS provides cloud server storage. And it also provides convenient memory management. Storage sense allows you to manage your files and the space they are using. It also possess Disk cleanup to delete unnecessary files. Disk management is a windows system utility that is mainly used for advanced storage tasks.
4. **Memory Management**: Windows provides both physical and virtual address space which allows up to 4 gigs of memory to run the applications. The memory management built-in as a system utility tool would be used to allow the application to access the database of images.
5. **Distributed Systems and Networks**: Distributed systems and networks can have common issues such as queuing repercussions as well as routing and congestion problems. However, these systems provide easy communication and coordination between one another. Some common problems in relation to using distributed system-- which can results in lagging computing performance and connection problems among individual users-- include independent failed components, absence of global clock, simultaneous computations of components
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 built-in anti-spyware solutions that keeps crooked or unwanted software from getting into the system. VPN service capabilities help to protect the client information and history from being used for malicious activities; RDP stream cipher that calls encryption protocols for smaller amounts of information, i.e. passwords, credit card numbers, social security numbers, etc. Alternatively, Windows 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. Windows defender is also part of windows security function.