

Draw it or Lose It

# **CS 230 Project Software Design Template**

Version 3.0

## Table of Contents

[**CS 230 Project Software Design Template** 1](#_heading=h.gjdgxs)

[**Table of Contents 2**](#_heading=h.30j0zll)

[**Document Revision History 2**](#_heading=h.3znysh7)

[**Executive Summary 3**](#_heading=h.2et92p0)

[**Requirements 3**](#_heading=h.tyjcwt)

[**Design Constraints 3**](#_heading=h.1t3h5sf)

[**System Architecture View 3**](#_heading=h.4d34og8)

[**Domain Model 3**](#_heading=h.2s8eyo1)

[**Evaluation 4**](#_heading=h.17dp8vu)

[**Recommendations 5**](#_heading=h.26in1rg)

## [Document Revision History](#_heading=h.lnxbz9)

| Version | Date | Author | Comments |
| --- | --- | --- | --- |
| 3.0 | 2/18/2023 | Ben Webster |  |

**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](#_heading=h.35nkun2)

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](#_heading=h.1ksv4uv)

* 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](#_heading=h.44sinio)

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](#_heading=h.2jxsxqh)

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](#_heading=h.z337ya)

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** | Flexible terminal commands to configure the server, access, or make changes.  Advantages It is upgradeable, it has various options for different web hosting requirements.  Disadvantages It is less preferred for web hosting services | The same goes for mac plus more cost-friendly  Advantages Security flaws are found before they become a threat, it is the most preferred choice for web hosting services  Disadvantages It is more difficult to use if the user does not have Linux experience. | 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 set in a single location rather than a mobile one. Specifications are better in other devices.  Characteristics: More popular, high portability.  Advantages Impacts more users, better compatibility, cost-effective  Disadvantages It is highly selective to various smart mobile devices Poor security |
| **Client Side** | Moderate expertise and time required. Cost similar to windows. | Maximum expertise and time required. Minimum cost. | Minimum expertise and time required. Cost similar 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** | When running languages on macs we can run swift, the more popular option. While mixing in nice tools like notepad++. Though Macs can run all languages. | Linux can work with visual studio, eclipse, along with notepad++ for a nice and easy-to-use tool. Along with numerous more languages and tools. | Easier to use than Linux but can run the same as it. So visual studio, eclipse to name a few of the many languages. With multiple tools notepad++ is simple to use | App creation using android and swift unlocks endless application potential. Both languages and software can be run on all three machines. |

## 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**: Having a game on multiple different platforms means that the game will need to be tailored to that specific one. However, with a Windows Operating Platform; many features can be implemented to work on all OS and ensure that the game is created on a well-balanced Operating Platform.
2. **Operating Systems Architectures**: Windows offers a simple, easy-to-use, platform with a simple GUI. Windows also has numerous software packages available in many different programming languages.
3. **Storage Management**: Included with Windows, there are simple configuration settings that allow for memory management. This allows you to scrutinize and manage files on your hard drive, along with how much space it takes up. Along with that, Windows offers cloud storage abilities that can provide developers with as much storage as needed.
4. **Memory Management**: For this game, you will need a database or library that has and can hold many pictures. The memory allocation allows for easy storage of pictures outside of the default picture folder. Windows does this especially well, including physical and virtual address space allowing up to four gigabytes of memory.
5. **Distributed Systems and Networks**: Distributed systems and networks can have common issues like queuing repercussions as well as routing and congestion problems. Many common problems occur from the relation to using the distributed system which can range from lag in computing performance to connection issues with multiple users.
6. **Security**: Windows operating platforms provide users 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 the administrative user(s). Windows also includes anti-spyware that comes built-in with a Windows device.