

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 | <8/12/21> | David Ferrante | Changes in descriptions pertaining to operating system development.  Revision of 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 is creating a team drawing game and are having an issue creating a vaiable setting. To set up this environment in java through eclipse I need to create instances for the teams and names to be unique to the players.

## [Design Constraints](#_2et92p0)

Web based constraints come in running executable jar files. As well as the multitude of screen sizes and shapes for a working display and easy use of entering team and player information as well as the drawing.

## [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)

With the Entity class being a base class holding common variables, the game, team, and player classes will be extensions of the entity class. The GameService class will also work with all of the extended classes to gather and use the unique id’s and names of the players. With the driver and singleton tester files as standalones to make sure everything is working properly.

****

## [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** | Built in terminal for access and alterations in the server.  Runs on a limited hardware options and missing items that would be on other operation platforms. | Similar to Mac but with an easier to navigate but limited terminal.  Powerful and stable servers for long term running. | Most software available for access and mutation of the server. Like Linux has good strong hosting capabilities. | Minimal access to server with weaknesses in being viewer friendly. Has a constantly moving server and may halt connections. Issues with server integrating all different mobile device type and systems. |
| **Client Side** | Well rounded with medium expertise, cost, and time required. | Cheapest but will require the most time and effort in understanding the operating system. | Lowest expertise and time required with moderate cost. Great for web site development as well as window’s app. | Time varies and updates can be viewed easily by others. Will require medium expertise. Something that will need to be taken into account is screen size for the ui, especially in a drawing game. |
| **Development Tools** | Eclipse, python, visual studio depending on how it is run. | Linux supports most programming languages, it may run into snags | Eclipse, python, visual studio as well as others. More options than any other os. Specializing in the C++ language but will run most anything. | Unity, jqueuery, sprout. IOS. For the most part mobile games will be coded in various Java languages as it is easy for the operating system to process. |

## 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**: The windows operating system would be the best option for creating and maintaining draw it or lose it. Windows is the most qualified as it is able to build the game for any of the platforms
2. **Operating Systems Architectures**: Windows NT created by Microsoft is doubled up with user mode and kernel mode. Allowing for access to multiple levels of the applications and hardware. With User Mode for regular running applications and the Kernel mode for accessing the more in-depth hardware.
3. **Storage Management**: Windows uses a dedicated disk management software including cloud accessible OneDrive to allocate memory. Along with the SSD this lets the resources for the application be stored and accessed easily without compromising on speed.
4. **Memory Management**: Windows has a large virtual address space accessible by all threads while also preventing corruption by crossing processes. Each of these can store up to 4 gigabytes which will be enough to store all the imaged that will be used. Using it in this fashion will also protect the game from crashing and corruption.
5. **Distributed Systems and Networks**: Requiring more effort but paying off in the end, creating a multi-faceted shared database that is network accessible will allow for expansion into other OS’s to establish cross-play will be best for expansion of Draw It or Lose it. Using this also supports the game for being used on multiple devices and with many different players at different locations at once.
6. **Security**: Microsoft has many secure data center facilities to protect volatile personal data, along with the private attributes and classes in the coding for draw it or lose it the players information will be secure from outside threats. Creating the game Microsoft will also support having user credentials to help secure the account from others and intrusions into the system.