

Draw It or Lose It

# **CS 230 Project Software Design Template**

Version 1.1

## 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 | 3/21/2021 | Ryan McFarland | Initial document creation. |
| 1.1 | 4/3/2021 | Ryan McFarland | Completed evaluation and recommendation. |

**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)

Draw It or Lose It is a current Android game developed by The Gaming Room. The game is loosely based on win, lose, or draw and players guess the shapes as they are rendered by the game. Drawing takes place progressively over a 30 second time frame. If the current team does not guess correctly once the 30 seconds have elapsed the other teams can offer a guess within a 15 second window. In order to begin the process of creating a web-based version of the game, the following requirements have been established: a game will have the ability to have one or more teams involved, each team will have multiple players assigned to it, game and team names must be unique to allow users to check whether a name is in use when choosing a team name, only one instance of the game can exist in memory at a time. Each instance of game, team, or player will have unique identifiers.

## [Design Constraints](#_2et92p0)

Constraints particular to a web-based game vs an android version are that users will be required to have an internet connection. At the time of writing, I am unsure if the android version enables offline play. Additionally, any device functionality used in the android version will need to be rethought as accessing the system camera, microphone, or other common peripherals in mobile devices will be much more involved on a web-based game.

## [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 all inherit attributes from the base class “Entity”. Inheritance allows for the reuse of code throughout classes efficiently. The main in program driver calls the testSingleton method which tests the effectiveness of the singleton pattern. Encapsulation protects the GameService() default constructor from being called outside of the object's own methods. The choice of creating classes makes the designs portable, the code can be used in other applications.



## [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.

| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| --- | --- | --- | --- | --- |
| **Server Side** | MacOS Server  More suitable for small network device control.  Server.app not regarded highly.  Low cost.  A dedicated network professional will be required.  Server.app is a $19.99 app that is not an OS in and of itself.  Research indicates that the current supported version stripped of functionality. | Many distributions to choose from, Ubuntu is popular.  Security guarantee.  Low cost.  Cloud support.  LDAP.  Enterprise support available.  An industry standard for servers so solutions will be available for most problems.  Command line based. | Windows Server.  Licensing fee.  Admin Center.  Windows Defender Advanced Threat Protection (ATP).  LDAP.  Cloud support.  Lower skill level required.  Comes with a graphical user interface. | Not used for hosting web-based software applications except in localized “maker” instances.  Cellular and WIFI communications expose data.  Not a viable candidate to host an application. |
| **Client Side** | Can use React Native to develop for Mac. | Can use React Native to develop for Linux. | Can use React Native to develop for Windows. | Can use React Native to develop for Mobile. |
| **Development Tools** | Visual Studio  Objective C  Swift  Java  Python  **React Native** | C/C++  Java  Python  Eclipse  Visual Studio  **React Native** | C/C++  Java  Python  Eclipse  Html5  CSS3  JavaScript  **React Native** | Android Studio  Kotlin  **React Native** |

## **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 recommend using a Linux distribution as an operating platform. The Ubuntu distribution is used widely and highly regarded as a go to solution to host applications and make them available to many platforms.
2. **Operating Systems Architectures**: The Linux operating system is composed of its applications like device drivers, libraries, daemons and its kernel space arranged in a layered method that simplifies the design and implementation. The kernel space of Linux is resistant to outside influence from rogue or malicious applications and over 1000 developers contribute to just the kernel.
3. **Storage Management**: I recommend redundant arrays of independent disks (RAIDs) in order to achieve a high level of performance and reliability. RAIDs provide high speed data transfer, and the redundancy means that data is safe in the case of disk failure.
4. **Memory Management**: Linux memory management subsystem is responsible for memory management in Linux. Virtual memory, demand paging, a layered internal structure, address space allocation are all taken care of by the memory management subsystem.
5. **Distributed Systems and Networks**: The cross-platform implementation of Draw It or Lose It will reside on servers situated among nodes to increase the chances of remaining operable in the case of local outages. With this in mind, as far as startup costs are concerned, it may be preferable to consider a cloud-based storage management solution to reduce the costs. In any case, the servers will respond to requests made by different platforms. The implementation should be done according to RESTful design principles. An increase in server nodes, done strategically, will increase the likelihood of remaining operable and help to spread traffic across locations so that any one server won’t be overwhelmed. Design should include an offline mode for mobile so that when a user is without internet access they can still use the app, perhaps in a limited fashion.
6. **Security**: User information should be protected by making use of the Linux server platforms security tools. Among the security tools offered on Linux are: OpenSSH which includes strong encryption, encrypted storage, package management that keeps software updated, system recovery, user accounts, file permissions, and several others. To take advantage of the platform's secure nature, any user verification and sign in should be done by the server and not by the client. Any traffic between the client and server should be encrypted. Employee and company logins, emails, and any software in use should be validated before use. Audits by third parties should be used to assess the worthiness of security measures.