

Draw It or Lose It

**CS 230 Project Software Design Template**

Version 3.0

# Table of Contents

[Table of Contents 1](#_Toc312943037)

[Document Revision History 1](#_Toc1907673007)

[Executive Summary 1](#_Toc1247188579)

[Requirements 1](#_Toc185754539)

[Design Constraints 1](#_Toc495247792)

[Domain Model 2](#_Toc946982813)

[Evaluation 2](#_Toc2085521266)

[Recommendations 3](#_Toc1669914739)

# Document Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Comments |
| 1.0 | 03/30/2025 | David McMurray | Wrote initial entries for all sections |
| 2.0 | 04/11/2025 | David McMurray | Added table of contents. Removed assignment instructions. Spelling and grammar corrections. Minor adjustments to evaluation and recommendations. |
| 3.0 | 4/27/2025 | David McMurray | Updated storage and memory management recommendations. |

# Executive Summary

The Gaming Room ("the client") wishes to develop a web-based version of their existing Android game, Draw It or Lose It, but the client's staff are not familiar with setting up a web development environment. Creative Technology Solutions ("CTS") will facilitate the client's development of the web-based game by advising on operating platforms and development tools and by writing initial scaffolding code to handle unique games and teams.

# Requirements

1. The product must be web-based.

2. Each Game will handle one or more Teams.

3. Each Team will have multiple players.

4. Game and Team names must be unique.

5. There must not be multiple instances of the same Game in memory.

# Design Constraints

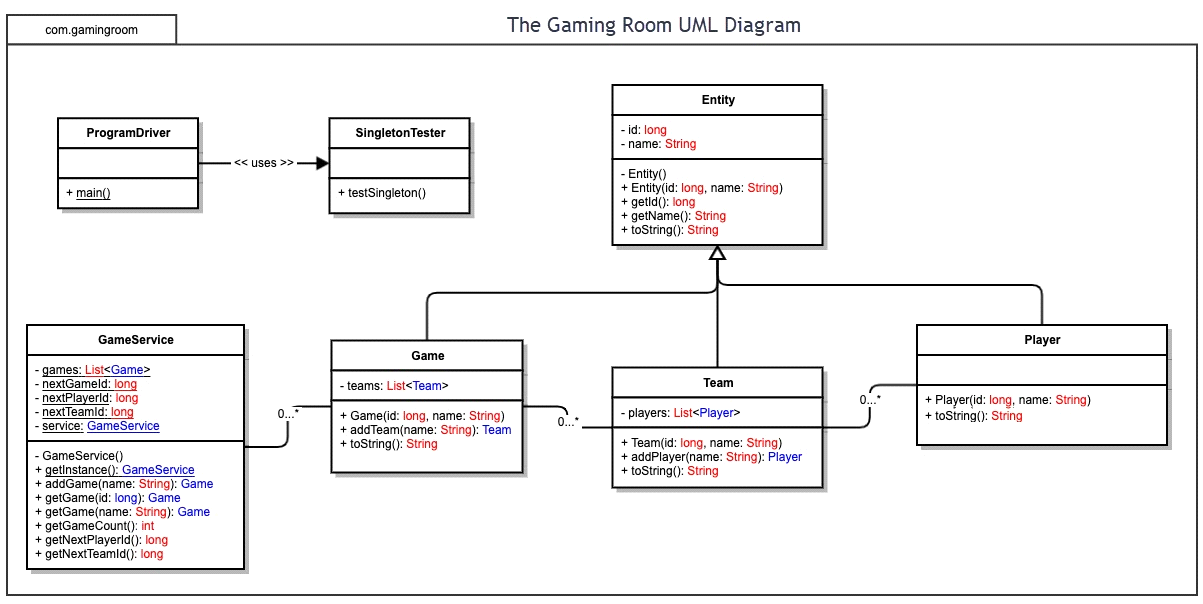
1. The solution will only implement the scaffolding described in the domain model. The client is well-equipped to re-implement the detailed functionality of the game. The client has requested aid in streamlining the development, not asked for CTS to port the full application.

2. The solution will not use advanced frameworks or libraries. Sticking to an implementation with minimal to no third-party dependencies avoids potential refactoring work in the event that the client discontinues use of that dependency in the future.

3. The solution will minimize how much data needs to be transferred to the client wherever possible. Unlike the native Android app stored on users' devices, web applications may need to be redownloaded each time they are used. The amount of local cache available for the images used in the game is also unpredictable when serving many different types of clients.

# Domain Model

The diagram below represents a system that supports Game, Team, and Player objects. These three objects represent important concepts used in Draw It or Lose It. Each of these three inherits from the Entity class, to avoid duplicate code for common data members. There is also a GameService class which ensures that there are no duplicate Games, Teams, or Players. To be accessible from everywhere in the program and avoid duplicate inconsistent records, the GameService is implemented as a singleton. There are also two classes for running the demonstration program, ProgramDriver and SingletonTester. By separating the big-picture code for the main function of the program from the code for the classes, the classes remain modular and reusable for other programs.



# Evaluation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| **Server Side** | Macs are an unconventional but viable option for hosting a server. They are generally powerful machines but offer limited low-level configuration. | Linux is a strong option for hosting a server. It can be loaded on many kinds of hardware and configured in great detail. However, it can be more difficult to use than Mac or Windows. | Windows is a strong option for hosting a server. It is easy to get Windows running. Windows is a middle ground between Mac and Linux in terms of breadth of supported hardware and depth of configurability. | Mobile devices are not suitable for hosting servers due to their limited storage space and CPU throughput. |
| **Client Side** | To maximally support Mac clients, it will be necessary to ensure that the game only uses browser features supported by Mac's default browser, Safari. Additionally, the UI will need to be adapted to larger screens. | Linux desktops are relatively uncommon and variable. To have the best chance of supporting them, the game should only use features supported by the most common browser in Linux distributions, Mozilla Firefox. Additionally, the UI will need to be adapted to larger screens. | To maximally support Windows clients, it will be necessary to ensure that the game only uses browser features supported by Windows' default browser, Microsoft Edge. Additionally, the UI will need to be adapted to larger screens. | The most common operating systems on mobile devices are iOS and Android. To maximally support them, it will be necessary to ensure all browser features the game uses are supported by Safari and Google Chrome. |
| **Development Tools** | Programming the website for the game will require the use of JavaScript, HTML, and CSS; this will be consistent across client platforms. No specific IDE is required. Acquiring a Mac machine is recommended for testing the website on Safari. | Programming the website for the game will require the use of JavaScript, HTML, and CSS; this will be consistent across client platforms. No specific IDE is required. Mozilla Firefox is cross-platform, so a Linux machine is not necessary. | Programming the website for the game will require the use of JavaScript, HTML, and CSS; this will be consistent across client platforms. Acquiring a Windows machine is recommended for testing the website on Microsoft Edge. | Programming the website for the game will require the use of JavaScript, HTML, and CSS; this will be consistent across client platforms. No specific IDE is required. Acquiring an iPhone is recommended for testing the website on mobile Safari. Presuming that The Gaming Room already has Android devices, keeping one is recommended for testing on mobile Google Chrome. |

# Recommendations

* **Operating Platform**: CDS recommends that the client uses Windows for servers and development.
* **Operating Systems Architectures**: Windows is a general-purpose safe first choice for servers, development machines, and more. This project does not call for the fine-grained control offered by Linux, and the client has not indicated that they are locked into the Mac ecosystem. The choice of development machine OS is flexible, but using the same type of machine as the one that you host from can ease the knowledge burden, compared to learning to work with two different ones.
* **Storage Management**: CDS recommends that the client continues with their existing storage management plan for user profiles and game image, preferably local to avoid cloud costs, with the caveat that images are stored in a compressed format, ready to be transferred to clients as-is.
* **Memory Management**: Windows, like most operating systems, offers functions for dynamically allocating and reclaiming memory as server use increases and decreases. This project does not call for special lower-level memory handling; using the server software’s programming language’s normal memory handling will suffice. CDS recommends that the server caches all compressed images in RAM to minimize demands to load for storage, as the image library is less than two gigabytes.
* **Distributed Systems and Networks**: To keep communication standard between different kinds of clients, CDS recommends that the clients only communicate to each other through the server, not directly to each other. Pre-sending the next few images, or sending all information required for a game at the beginning if feasible, can mitigate the "lag" users experience in the event that their connection is interrupted, or bandwidth is reduced momentarily.
* **Security**: As the client is already operating this application for a specific platform, CDS recommends that the client extend their existing authorization and security practices to include the new web-based game.