

Draw It or Lose It

# **CS 230 Project Software Design**

Version 3.0

## 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 | 01/23/2022 | Andreas Galatis | Updated Executive Summary, Design Constraints, and Domain Model |
| 2.0 | 02/06/2022 | Andreas Galatis | Updated the Evaluation of Development Requirements (Server/Client/Tools) |
| 3.0 | 02/20/2022 | Andreas Galatis | Updated the Recommendations section |

**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)

This document provides a design overview of the gaming software being built for The Gaming Room. The game is a rendition of the 1980’s television game *Win, Lose, or Draw*. A game consists of four rounds of play lasting one minute each. Drawings are rendered at a steady rate and are fully complete at the 30-second mark. If the team does not guess the puzzle before time expires, the remaining teams have an opportunity to offer one guess each to solve the puzzle with a 15-second time limit.

The technical requirements for this game application will include:

* The ability to have one or more teams involved
* Each team to have multiple players assigned to it
* Game and Team names must be unique
* Only one instance of the game can exist in memory at any given time

## [Design Constraints](#_2et92p0)

The design constraints that will have an impact on the development of this application in a web-based distributed environment include:

* Platform: Using multiple platforms will have several impactors factors in gameplay application such as screen sizes, computing power, and input devices.
* Memory limitations: Many web-based games are played on mobile devices which will have significant memory limitations, as well as framerates on browsers.
* Business constraints: Some business constraints such as intellectual property restrictions will also affect design elements
* Thematic constraints: Preferences in themes based on what’s trending and what’s normative will affect user experience design and implementation.
* Time Constraints: Time allotted for the development of the game application will have a direct impact on its complexity in level implementation and scripting.

## [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)

Some of the OPP principles depicted in the UML class diagram include inheritance. By creating a base Entity class, we can minimize the redundancy in code by introducing several functions such as the getId() and getName() methods and a super constructor that can be inherited by child classes Game, Team, and Player. This also enhances reusability options.

Additionally, in this game application’s UML, we see a Multiplicity Association between the classes depicting the cardinality of the classes in relation to one another. For example, there are multiple games that can be created from a single instance of Game Service, and multiple teams from the game class, as well as multiple players from the team class. This makes it so that there only one instance of the game in memory at once, but there can be multiple games, players and teams stored in memory at the same time.

Finally, there is a Directed Association between the Program Driver class and the Singleton Tester method. This is an association that is only navigable in one direction. So, in this case, the Program Driver has a Singleton Tester for the purpose of testing the Singleton pattern used on Game Service to ensure that one and only one instance of Game Service does indeed exist in memory.

**"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)

| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| --- | --- | --- | --- | --- |
| **Server Side** | An open-source software type offering a server-based deployment method in the form of Apache, the most widely used HTTP server on the Internet. Scripting using Perl, Ruby, and Python.  Advantages are it can host multiple websites on a single server, using SSL and realm-based authentication  to secure network transactions and control access to web content. Apache is preconfigured with default settings, so deployment  is as simple as selecting Start Service in the Server Admin utility  Disadvantages are compatibility and restricted hardware upgrades with limited maintainability. | An open-source software type offering a server-based deployment method in the form of Apache and Nginx. Uses Perl, PHP, Python, and Ruby as script languages.  Advantages are it’s free of charge and supports cooperative work, without users being able to damage the program’s core. Rarely experiences security errors and has few demands on hardware.  Disadvantages are its operation is complex as well as the update process. Several professional programs do not work on Linux. | A proprietary software type offering a server-based deployment method in the form of Microsoft IIS, using VBScript and ASP.NET as script languages.  Advantages are beginner friendly, supports large number of third-party applications with easy and optional automized system updates.  Disadvantages are high licensing costs, with increase with each user. Vulnerable to malware and often security-related errors. Not best suited for multi-user systems | Open-sourced software, most common are HTTP servers or servlet containers such as Jetty, Tomcat, Glassfish, and Resin.  Advantages are compatibility with various types of mobile devices. Broader reach due to multi-device support that responsive web design provides across various platforms.  Disadvantages are differences in user experience for multiple platforms. Limited offline functionality. |
| **Client Side** | Family of Macintosh operating systems, developed by Apple Inc.  Advantages are ease of use for the non-technical favoring content creation. Needs Almost no spyware or virus applications. More powerful, and more stable.  Disadvantages are it’s more expensive. Exclusive support can only go to Apple. Less hardware choices. | Developed and marketed by Microsoft.  Advantages are more than 80% of PC users run on Windows. Ample support base, and hardware and driver availability. Most widely supported operating system for games.  Disadvantages are it’s the most prone to spyware and virus applications. Poor security and general instability. | Family of open-source Unix-like operating systems based on the Linux kernel  Advantages are it runs on just about any hardware with the largest driver support of any system. Everything is free. Full access to the free open-source library of software.  Disadvantages are slow innovation with limited vendor support, as well as limited support for proprietary applications. | Responsible for the logic side and implemented using JavaScript.  Advantages are convenience as content load faster and are easier to use. Allows for personalization to use preferences. Can be used offline.  Disadvantages are compatibility must be OS specific, Android, iOS or Windows. Support often takes more time and costs more. |
| **Development Tools** | Apache 1.3 and 2.01 with HTTP 1.1 support  Support for virtual hosting, including multiple  IP addresses and virtual domains  Encrypted data transport with support for  SSL and TLS protocols  WebDAV support for collaborative content  Publishing.  XCode: This is the IDE used to create macOS apps. | Most of the Linux kernel code is written using the GNU extensions of GCC to the standard C programming language.  LinK+ IDE is a simple IDE for Linux Kernel Developers. It is based on Eclipse IDE customized for Linux kernel programming | The Windows SDK provides access to all the APIs and development features exposed by the Windows OS.  Uses Visual Studio IDE.  For app development, will need he Windows App SDK provides a unified set of APIs and tools that can be used in a consistent way by any desktop app on Windows 11 and down level to Windows 10, version 1809. | Android is written in Java (UI), C (core), C++ and others and developed mostly by Google. Its OP is based on a modified version of the Linux kernel.  Swift is the powerful and intuitive programming language for iOS. Can be developed using XCode.  Windows app is written in C++, using Visual Studio |

## 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 recommended operating server platform for The Gaming Room to expand to other gaming environments is the Windows OS. Windows is the most unrestricted operating system that allows various software integrations with third-party applications, resulting in a cost-efficient solution. The Mac OS, on the other hand, has several compatibility limitations, while Linux’s updating process is complex and not very user-friendly. Additionally, Windows’ support availability is widespread, with consistent and straightforward updates, making it the best and consistent server to provide fast and efficient upgrades.
2. **Operating Systems Architectures**: Consisting of low-level operating system functions, the Windows architecture comes in user mode and kernel mode. The Windows kernel uses thread scheduling with interrupt and exception dispatching to perform maintenance before transferring execution over to the hardware drivers. Additionally, the Windows OS utilizes multiprocessor synchronization which aides in concurrent program execution, as well as read/write sharing of data.
3. **Storage Management**: An appropriate storage management system to be used in Windows will be one that utilizes the Windows storage management API. This storage management provider can be used to handle a large scope of storage configurations. The Windows Storage Management API provides extensive storage provisioning and administration capabilities. Additionally, storage management is fully scriptable, and administrators can manage it remotely.
4. **Memory Management**: Windows implements virtual memory, which can have the semblance of providing unlimited memory. Virtual memory also provides memory protection and will also help maximize speed for game play since only a particular segment of the program is required for the execution of the application. Windows memory management specifications consist of each process on 32-bit Microsoft Windows having its own virtual address space that enables addressing up to 4 gigabytes of memory. Each process on 64-bit Windows has a virtual address space of 8 terabytes. All threads of a process can access its virtual address space.
5. **Distributed Systems and Networks**: For *Draw It or Lose It* to communicate between various platforms, a server needs to execute the game, followed by a transmission over a network to the client. For this, Windows networking functions enables the implementation of network capabilities into the gaming application. Windows does not, however, make allowances for a particular network provider or network implementation. This is achieved through wired access deployment which requires server certificates. A server certificate is a digital document that is commonly used for authentication and to help secure information on open networks.
6. **Security**: Windows Security is built-in to Windows and includes an antivirus program called Microsoft Defender Antivirus. Installing a third-party antivirus will disable Microsoft Defender, therefore, maintaining an update version of this program Is vital for security. Also, periodic disk checks and minimizing administrative login to the system is recommended to keep an as-privileged access to the hard drive.