

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 | 03/09/2022 | Brandie Sheppard | Managed the requirements and constraints regarding the app creation based on the company’s existing website. |
| 2.0 | 3/18/2022 | Brandie Sheppard | The Gaming Room would like to expand its reach beyond the web and sell across all platforms to increase its cliental. We have improved the details regarding the project constraints as well as the different requirements needing to be changed for the program to be successful. |
| 3.0 | 04/15/2022 | Brandie Sheppard | Revised and added additional developer tools, and made some recommendations based on **Operating Platform, Operating Systems Architectures, Storage Management, Memory Management, Distributed Systems and Networks, and Security.** |

**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 paragraph response covering the indicated information.

## [Executive Summary](#_sbfa50wo7nsh)

The gaming Room wants to develop a web-based game resembling the 80s television game “Win, Lose, or Draw,” where teams compete to guess what is being drawn. The purpose of the game is multiple teams with multiple players on each team to play four rounds. Each round consists of 1 min. Drawings are then constructed at a steady rate and are fully illustrated after 30 seconds. If the team does not provide the correct answer before the time expires, the opposing teams get an opportunity to offer one guess each to solve the puzzle within 15 secs. The current app is available for Android only and the client would like to expand to multiple platforms to sell the game.

## [Design Constraints](#_2et92p0)

* The game must consist of one or more teams
* Each team must consist of multiple players
* Game and team names must be verified if already in use to keep teams and names unique
* Only one instance of the game can exist in memory at any given time.

## [System Architecture View](#_ilbxbyevv6b6)

These are the requirements needed to follow while writing the code and software. While this is only the game aspect, we still need to look at application development. The Gaming Room would like this to run on all devices. This means we already have it on android but need to work it into another mobile device. Along with machines like Windows, Linux, and Apple. To do this we will need to find a way to either re-write the code in swift for (Apple devices) or come up with a way to use existing code to be run on other devices by inheriting other languages. Like when we use multiple computer languages together to make stronger code.

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 entity creates a relationship between the game, the team, and the player class, meaning they get their information from the entity. Each class shares common references, like the name and ID, making the entity a “Superclass.” When we use a UML, we call it aggregation. It’s an instance of one class and has a reference to another class. When we look at the diagram we see “GameService” has a reference to “Games,” “Games has a reference to “Teams,” and “Teams has a reference to “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](#_2o15spng8stw)

Using your experience to evaluate the characteristics, advantages, and weaknesses of each operating platform (Linux, Mac, and Windows) as well as a mobile device, 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 paragraph response covering the indicated information.

| **Development Requirements** | **Mac** | **Linux** | **Windows** | | **Mobile Devices** |
| --- | --- | --- | --- | --- | --- |
| **Server Side** | **Advantages**  It’s upgradeable  it has various options for different web hosting requirements  Flexible terminal commands to configure the server, access, or make changes.  **Disadvantages**  Not a good choice for gaming  Not very many touch screen functions | **Advantages**  Security flaws are caught before they become an issue  Comes completely free and open source  **Disadvantages**  Linux in general is more tech-savvy than the others.  Fewer options for commercial apps. | **Advantages**  High resource requirements  fewer load times  high comfortability  More software available compared to other OS  **Disadvantages**  Easy virus susceptibility  Poor tech support | | **Advantages**  Have a wider reach, and better compatibility  cost-effective  It's better if the server is immobile and can be tracked in a single place  **Disadvantages**   It is highly selective to various smart mobile devices  Poor security |
| **Client-Side** | Moderate expertise and time required. Cost is like windows. | Maximum expertise and time required. Minimum cost. | Not as secure as Mac or Linux | | Provides flexibility to clients or even developers to see updates at any place. Slightly more difficult to implement than other devices. |
| **Development Tools** | Macs can run multiple languages including HTML, CSS, and JavaScript while supporting libraries to support the frontend and general-purpose languages. | Linux can work with Visual Studio, eclipse, as well as notepad++.  languages including HTML, CSS, and JavaScript, while supporting libraries to support the frontend and general-purpose languages. | Easier to use than Linux but can run the same as it. So visual studio, eclipse to name a few of the many languages. And with multiple tools notepad++ is a simple to use tool. Languages consist of but are not limited to HTML/CSS/JavaScript while supporting libraries to support the frontend and general-purpose languages. These can be Java, Python, PHP, and Ruby. | You can create countless apps using android and swift. Both languages and software can be run on all three machines. Languages consist of but are not limited to HTML/CSS/JavaScript while supporting libraries to support the frontend and general-purpose languages. These can be Java, Python, PHP, and Ruby. | |
| **Universal tools that are used for and across multiple platforms:**    RAD Studio is used across all platforms and enables you to write your code once and use it everywhere. Connect to over 20 databases natively with high-speed direct access.  SmartBear Collaborator is a peer code and document review tool for developers. It allows you to review source code, ensures proof of reviews with electronic signatures with detailed reports, comes with Microsoft Word, Excel, PowerPoint, and Visio, and much more, all in one quick tool.  ATOM is an all-around text editor. It’s open-source and completely free and works across multiple operating systems while easily opening one project or multiple projects open in one window. You can also find, preview, or replace any text type in a file or across the entire project.  Visual Online (Visual Studio Online) is a collection of services. It is fast and easy to plan, build and ship software across a variety of platforms. It is one of the best tools for software developers that allows organizations to create the perfect development environment. It allows you to track and manage all ideas on Kanban or scrum boards with agile tools, improve code quality, and build, manage, and share software components. It also offers a centralized control system with free private repositories, | | | | | |

**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 would recommend The Gaming Room starts on windows devices as it has more software available along with minimum expertise and cost to get projects going. You also won’t run into a shortage of IDEs to work with.
2. **Operating Systems Architectures**: Windows provides services used by all Windows-based applications that enable applications to show a Graphical User Interface (GUI) while accessing system resources and much more. These applications also refer to Graphics and Multimedia, messaging, and web services. These services can be used using a user account or a server specifically.
3. **Storage Management**:  Windows 10 comes with a nice feature called storage sense. This allows you to scrutinize and manage files on your hard drive and how much space it takes up. Other features include being able to choose to save locations for apps making them easier to find. And just like other dives, you can also use the cloud to save data. The built-in storage system allows for easy file creation and placement for large projects, so they won’t get lost or carelessly deleted.
4. **Memory Management**: While creating this game you will need to create a database or library with lots of pictures. The memory allocation allows for easy storage of pictures outside of the default picture folder. This allows you to keep your whole project together in a more secure area on your computer. This includes when you’re working with your IDE and opening files from it to create the game.
5. **Distributed Systems and Networks** Because each operating system is different I investigated ways to publish the game to run on all dives. I found Develop 4 which enables cross-platform game creation. It’s an IDE that can be run on any device. Once the game is created you can simply export the game file to the web, iOS, Android, and many more options that will allow cross-play. This will help with dependencies. To prevent other problems like outages or connectivity, the company will need to ensure its servers are strong enough to support large player volumes and backup power for power outages.
6. **Security**: Windows comes with built-in security protection software. Though to secure user data and information it would be recommended to use another source. Though if we are talking about what is on the machine windows comes pre-equipped with protection. This system scan for malware (malicious software), viruses, and security threats. This all happens in real-time, and because threats change the system updates automatically to keep the system and user information safe