

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  1.1  1.2 | 05/19/2022  06/01/2022  06/19/2022 | Santosh Gayak  Santosh Gayak  Santosh Gayak | Begin writing the template  Added evaluation  Added 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)

Creative Technology Solutions provide technology solution to the clients. The Game Room is a new client who wants to build a web-based game which can be played in different platforms. Currently, the game is available only on the Android app.

In the game, computer renders the image from stock drawings at a steady rate and fully complete in thirty seconds. The game consists of has four rounds with each round lasting for one minute. So, if a team does not answer the puzzle, remaining team will get the turn and has fifteen seconds to guess.

## [Design Constraints](#_2et92p0)

Software requirement:

* The game can have one or more teams for a play.
* There should be multiple players in a team.
* Name of Game and Team must be distinct.
* At a given time, only one instance of the game can exist in the memory.
* The web-based game must be multi-platform friendly.

The game already exists in the android platform. The most popular platforms are MacOS, iOS, Linux, and Windows. The are various cross platform game development like Unreal engine and Unity. When using such Cross-platform game development software, it saves tremendous amount of time because only a singe programming code base can serve for different platforms, ultimately saving costs of build the 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 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.**

In the UML diagram above, ProgramDriver is the main function of the program and uses SingletonTester

Entity class is the base class and Game, Team and Player inherit from the Entity class. Each of the derived class will share the variable and methods from Entity class and have their own methods.

Team is associated with Game and Player is associate with Team. This means that in a game, it can have zero to many teams and in the team, it can have zero to many Players. GameService is associated Game with a relationship of zero to many.

## [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** | Mac web hosting can be very secured. Apple releases updates and upgraded versions of the server operating system.  It can be costly to use Mac servers. | Linux is Cost effective. It is Open source. It is secure comparing to other servers and very easy to use. It provides different interface choices and operates faster. | Windows provide high security and reliability.  Its quite costly use windows servers as the software are proprietary.  Expertise required is lower as it is very popular. | It provides Portability and is  cost effective.  It provides high performance  There are high chances of security threat and also is very low reliability. |
| **Client Side** | It is most costly software and hardware.  It is most secure,  Hardware hardly fails.  Moderate expertise is required. | It is Cost effective since the open-source software is free.  Linus need high expertise and therefore require more time. | It is quite costly for setting up hardware and software. Expertise required is minimum. | Updates can be notified anywhere. |
| **Development Tools** | Swift and C# is mostly used in Mac. IDE includes XCode, AppCode etc. | HTML, CSS, java script, Java python, PHP, C , C++ are relevant programming languages to build the software for deploying on Linux. Visual basic and Eclipse are few of many IDE. | C++, C, Java, PHP, PowerShell et are some of many programming languages.  Linux. Visual basic, Eclipse is few of many IDEs. | Swift, java, html, CSS, Java script etc. are relevant programming languages. |

## 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**:

Windows would be my recommendation because it requires minimum expertise and is quite cost effective. There are many IDEs available to use.

1. **Operating Systems Architectures**:

Windows has 32 bit and 64-bit architectures. It has kernel mode and user mode. It provides GUI interface.

1. **Storage Management**:

Windows Storage Management manages various storage configurations from internal memory divide to external storage device. It is easy to create, move, delete the files.

1. **Memory Management**:

Memory management is a high priority in this game because it needs to store plethora of images to display during the game. The windows memory management controls memory and the process during the game is transferred to execution disk.

1. **Distributed Systems and Networks**:

The game contains multiple teams and players who will use different platforms to communicate during the game. Distributed systems and network play important role in communication among the platforms. StackPath is provides infrastructure for connecting various components of a system. The client’s requirement of cross platform play can be achieved using it.

1. **Security**:

To provide users with high level of security, windows provide so many software and keep updating or upgrading them