

<Name-of-Software-Application>

# **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 | <mm/dd/yy> | <Your-Name> | <Brief description of changes in this revision> |

**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)

The Gaming Room, a video game manufacturer has contracted Creative Technology Solutions (CTS) as their consultants in the development of their new game Draw It or Lose. The game is currently available on the Android OS. The staff at the gaming company do not know how they can set up the gaming environment. The consultant is mandated with streamlining the gaming environment. The game should have capabilities such as the ability to have one or more teams involved. Each one of the teams will have multiple players assigned to it. The name of the game and the teams 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 any given time. It is proposed that this can be accomplished by creating unique identifiers for each instance of a game, team, or player. This document outlines some of the basic requirements of the game.

## [Design Constraints](#_2et92p0)

The design constraints in the development of the game include:

1. Theme: This will be the most significant limitation to the game. It will mainly be informed by the message that the developer wishes to communicate
2. Time: The timeline provided by the gaming company will be a major strain. Therefore, the plan will be developed to fit this timeline
3. Team: To ensure that the game is developed in time, there will be an additional team on the fly. This will help to ensure that the strength and weaknesses of the members are managed.
4. Tools: This mainly includes the software required to develop the game. The choice of the software will mainly be informed by its use and the budget.

## [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 game entity has an id, and name. It is comprised of a game service, game team and player. The game service is comprised of the current game, the next game, next player and the next team. The game is defined by the list of teams. The team has a list of players. The player entity has the player id. The main relationship evident in the model is inheritance. The derived classes tend to use the characteristics of the existing classes.

****

## [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 provides stability and high performance for server side hosting.  It is relatively easy to install, configure, and set up MAC OS servers for hosting web applications.  MAC is, however, expensive when purchasing server licenses. | Linux provides more stability as well as enhanced performance and speed for server hosting.  Linux offers more features for server side hosting for addressing increasing demands.  Linux is also cheaper in terms of server licensing. | Windows has stability issues and slow performance in server side hosting of web apps.  Windows server hosing is expensive and applications are limited.  More resources are required in addition to often downtime for servers hosting the web apps. | Mobile devices have stability issues in server side hosting.  Mobile devices are limited in terms of performance and applications that can be hosted.  Mobile devices are relatively cheap in server licensing. |
| **Client Side** | Mac supports multitasking making it easy and simple to support multiple clients without wasting time.  Mac is costly in terms of client side applications development.  Mac does not require a lot of expertise and offers an intuitive and simple to use interface. | Linux is free and open source making it efficient for supporting multiple clients.  Easy to install and run, in turn, Linux is not time consuming.  Linux requires prior knowledge and expertise to support multiple clients. | Windows provides free access to IDEs and is widely available.  Windows is a lot cheaper for client side software development.  Expertise in Windows is relatively easy for software development. | Mobile devices take time when developing client side software applications.  Mobile devices also cost a bit more when supporting multiple types of clients.  Mobile devices require prior knowledge and expertise for client side application development. |
| **Development Tools** | * Swift * Netbeans * Eclipse   Solar2D | WebStorm for software development  Brackets for live HTML, CSS and JavaScript coding  Komodo  GNU Emacs | Java for software development  Python  CSS  PHP  Ruby | HTML5  Ruby  Kotlin  C# for software development |

## 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: Linux operating system is more appropriate for expanding the game to other computing environments.
2. Operating Systems Architecture: Linux operating system architecture comprises of a kernel, which is the central part of the OS and controls all major processes. The architecture of Linux also comprises of a system library, which provides special functions; as well as a shell and system utility for performing specialized tasks and processes.
3. Storage Management: In Linux operating system, the appropriate storage management system to be used is the Logical Volume Manager (LVM) for creating pools of storage. This will make it possible to manage storage for the Draw It or Lose It software.
4. Memory Management: Linux uses two memory management techniques. The first is segmenting for splitting memory address range. The second is paging for mapping address space to physical memory.
5. Distributed Systems and Networks: Linux offers a distributed inter-process communication (DIPC) feature for building and programming multiple and distributed systems and networks.
6. Security: Linux offers numerous protection and security capabilities, including encrypted storage, file permissions, user accounts, secure remote access, and system recovery.

References

Edge, C. & Smith, W. (2015). *Enterprise Mac Administrators Guide*. Apress Publications.

Fox, R. (2014). *Linux with Operating System Concepts*. CRC Press.

Mali, L. (2017). *Mastering in Windows 10 Operating System Volume I and Volume II: Window 10 Apps, Control Panel, Registry, Services, Tips & Tricks & Group Policy.* Notion Press.