

The Gaming Room

# **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 |
| --- | --- | --- | --- |
| 3.0 | 6/18/20 | Rushit Thakker | Third edit and final draft of the project. |

## [Executive Summary](#_sbfa50wo7nsh)

The software game “Win, Lose or Draw“ is designed based on 1980 television show game with the same name. The goal of the software is to emulate the game. This leads to some constrains that arise to the design of the game some to due to software itself. One specific problem presented in gaming room is to create a unique gaming instance for multiple teams with unique names contained with unique individuals. The solution to this can easily be engineered using existing software design principles. A singleton pattern can help ensure single instance of game and static instance of team and individuals can ensure unique individuals.

## [Design Constraints](#_2et92p0)

Detailed list of constrains and solutions.

* The game must be web based app
* The software should be able to add multiple teams. Each with unique name for team and players.
* Only one instance of the game must exist.
* One instance problem can be resolved by using singleton pattern.
* A list can be created to have the ability to add new teams and members. Each addition could be checked before entry to make sure it’s unique.

## [System Architecture View](#_ilbxbyevv6b6)

TBD.

## [Domain Model](#_8h2ehzxfam4o)

Entity class is the abstract class that defines some of the shared properties between game, team, and player. Game, team, and player then extend entity to use those shared properties. ‘GameService’ is contains a list of games.

Singleton is executed by program driver to allow it to test validity of singleton pattern.

****

## [Evaluation](#_2o15spng8stw)

Evaluation of various platforms their advantages and disadvantages in the context of game room development.

| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| --- | --- | --- | --- | --- |
| **Server Side** | Mac can be particularly good for server-side development & operations because it’s based on UNIX. Though mac | Linux has be the dominant platform for server side development. From efficient memory use to crash free operations for years Linux is very reliable for long term online service. | Windows has its own backend eco system. That has improved over time, but it remains a closed source proprietary system. | Mobile devices are generally not used as severs but the Linux architecture does allow for selective possibility. |
| **Client Side** | Client side for mac requires can be developed with relative ease with the help of standard front-end tools. | Linux has many distros using various UI’s. The best solution would be to keep the web app accessible through browser. | Clint side for windows also has a lot of support like from Microsoft and has plenty of libraries and development tool to make it easier. | Mobile development has been hot for past decade. This has led to mobile first development for many entities. |
| **Development Tools** | Mac has XCode for front end development. XCode is one tool for all development needs and is highly customizable for any development needs for any apple product. | The best tool currently in open source community is VS code. It works on any platform and it very versatile for any development. | Windows has its own proprietary version of visual studio it supports vast array of features including tools for professional testing. | Very little development takes place on mobile platforms. But some apps can be used for small edits. |

## 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 best severs side Operating system for Gaming room would be Linux. While it a professional version like CentOS or Open Suse is recommended any distro would be sufficient to meet the requirements.
2. **Operating Systems Architectures**: Linux is very customizable and has very flexible architecture that can be configured to any requirement. System calls for most Linux kernels are similar, enable easier server deployment.
3. **Storage Management**: Linux kernel is compatible with most of storage formats and systems. Standard prebuild editions use Journaling file system. But user can change it according to requirement.
4. **Memory Management**: Linux has one of the best memory management of any operating system. In addition to virtual memory like other operating systems Linux also has swap partition that acts as another cache for better data availability.
5. **Distributed Systems and Networks**: Unix family was created so that computers can communicate with one another. It provides fine grained control of interfaces and ports with specific control policies.
6. **Security**: Due to the level of user access and the open nature of the OS a well configured and timely updated system would be substantially more secure. One of the key factors in human awareness. Most companies have dedicated security teams that are responsible for maintaining security.