

<Draw It or Lose It>

# **CS 230 Project Software Design Template**

Version 2.0

## 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 | 04/22/2021 | Antonio Bethune | Product Backlog for Game Requirements |

## [Executive Summary](#_sbfa50wo7nsh)

The task at hand is to create an interface for our web-based game that allows for multiple teams and each team to have multiple players, with one unique game running for these participants. The challenge is to appropriately assign the players to a team and then assign those teams to a game that will then be rendered on a “server” for that game to be played.

## [Design Constraints](#_2et92p0)

Some of the design constraints when it comes to a web-based game is the way we implement how people will join a game and thereby get on a correct team and be identified with it throughout the iteration of the game. Also, we must make sure that each part of the game has unique identifiers from each player, team, and game itself. There needs to be a way to test this to make sure no two identifiers are the same.

## [System Architecture View](#_ilbxbyevv6b6)

## [Domain Model](#_8h2ehzxfam4o)

The UML diagram below shows how we map out unique identifiers and names for the 3 main parts of the game which are the Players, Teams and Games themselves. They each have their own class and inherit from an Entity class since they share the similar characteristics of id and name. Each team can have many players and each Game can have many teams, but they do not go back (i.e., no team has many games). There is also a GameService class that is made to handle all the games being played and it can hold many different games. All the program is running through the ProgramDriver with a SingletonTester running in the background to make sure there are no duplicated of any game running at the same time.

****

## [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** | Can be used for server side running using tools such as Apache, PHP, MySQL, or PostgreSQL. Various computers and machines can run the server side on a linked network allowing for expandability as well. | Can be used for server side running using tools such as Apache, PHP, MySQL, or PostgreSQL. Various computers and machines can run the server side on a linked network allowing for expandability as well. | Can be used for server side running using tools such as Apache, PHP, MySQL, or PostgreSQL. Various computers and machines can run the server side on a linked network allowing for expandability as well. | Cannot really be used to host multiple devices as a server. Do not recommend mobile devices for server-side development |
| **Client Side** | For a web-based application, you could use a variety of client-side tools that would make a unique web interface. You are probably going to need a 2-6 developers working in teams to build the UI/UX, the actual functionality and then the back end is separate.  With a Mac you can also make it for the apple store and then have the requirements for macOS met that way users can download it and not play in a web window. | <For a web-based application, you could use a variety of client-side tools that would make a unique web interface. You are probably going to need a 2-6 developers working in teams to build the UI/UX, the actual functionality and then the back end is separate | For a web-based application, you could use a variety of client-side tools that would make a unique web interface. You are probably going to need a 2-6 developers working in teams to build the UI/UX, the actual functionality and then the back end is separate.  With windows you can make it for the android market and have people download it to the computers through an emulator like Blue Stacks | For a web-based application, you could use a variety of client-side tools that would make a unique web interface. You are probably going to need a 2-6 developers working in teams to build the UI/UX, the actual functionality and then the back end is separate. For a phone you would be building for either android or apple markets so that is a mobile specific you would need to know in advanced |
| **Development Tools** | XCode would probably be the best and most unique IDE for Mac. Developing in either C, C++, or Swift would be a Mac specific requirement, but other IDEs are available that are not native to Mac. | The best bet would be to use java or c++ for most of the interface and then use python to connect with the back end. If you want an all in one for a web application, you can even go with ruby on rails | The best bet would be to use java or c++ for most of the interface and then use python to connect with the back end. If you want an all in one for a web application, you can even go with ruby on rails | You would not really be able to develop on your mobile device effectively |

## 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 Operating platform would be to use Mac as it can reach a wider range of development tools, so it will not hinder the programmers during development once a language is picked. Also using an apple-based tool will make it easier and more streamlined when packaging out the actual application. Since the apple store is easier to push to from another apple product and android is robust and suitable to be pushed out to on any OS (due to all the various packages), it would be beneficial and more cost effective to use mac OS as the operating platform.
2. **Operating Systems Architectures**: I would go with Ruby on Rails approach if you wanted everything done through one language for client and server side, otherwise I would use java/c++ with python as the language for server and client-side communication. When utilizing ruby on rails, the main language you would be interfacing is ruby. This is known as one of the easiest high-level languages to develop in especially for any kind of web-based application that does not really use flash. It is easily integrated with python for server-side management and has a ton of support in the open-source world. Other languages such as java or c++ are robust enough and have their own object-oriented structure that might be good for the development of the game, but the integration of the web can be more cumbersome with these languages.
3. **Storage Management**: Any of the architectures can use MySQL/PostgreSQL for data storage. MySQL is probably the most popular storage management system that is easily integrated with the python programming language. The user interface is easy to manage and does not require too much code knowledge to understand, allowing other non-developers to give input on the storage management if needed. There is also PostgreSQL which operates similarly but might have a user interface that is more complex. This can be good in terms of being more secure and not allowing non developers to go inside and break anything.
4. **Memory Management**: You can also use a mac for memory management, just have a dedicated Mac for web storage such as a mac mini. There are several different options that can be used for memory, but when working in the iOS family for the development I believe it would make sense to also utilize an iOS device for memory as well. This can allow easier cross functionality between systems and there is a certain level of quality built into apple products. Another big thing with memory is that you want it to move fast s to not hinder the user and using systems from the same company can help with the processing speed for different devices especially that of memory.
5. **Distributed Systems and Networks**: Considering the application is web based, any device that has access to the web will be able to use it from computers to mobile devices. Also, since we developed the actual application on a mac, the distribution to the iOS network will take less time since the apple store certificifications are easier to get on an iOS device. Other stores that we would place the finished product in would be the android market which can be accessed through google and that is also an easy store to get approval on no matter what device. This also means that if we get it within both large markets that we just must make sure that it meets the specs for computers as well and then we can have the option enabled for users to download the game to their computers through withretr the apple store on their macs or throught the google play store on other computers and use an emulator to play.
6. **Security**: For security purposes I would suggest that you get multiple testers to white hat hack for any vulnerabilities and use multiple web masked and VPN’s during games to hide players IP addresses. There are also various content managing sites you can use, but I would recommend that you have your dedicated servers being monitored for any form of malware and that there are daily security checks in place so that, if need be, the gaming site can be shut down and checked if any security breach would happen.