

Win, Lose or Draw

# **CS 230 Project Software Design Template**

Version 1.1

## Table of Contents

[**CS 230 Project Software Design Template**](#_l6ti7uoag22u)1

[**Table of Contents**](#_30j0zll)2

[**Document Revision History**](#_grjogdjh5fi8)2

[**Executive Summary**](#_sbfa50wo7nsh)2

[**Design Constraints**](#_2et92p0)2

[**System Architecture View**](#_ilbxbyevv6b6)2

[**Domain Model**](#_8h2ehzxfam4o)2

[**Evaluation**](#_2o15spng8stw)2

[**Recommendations**](#_m8aleynsvzvc)2

## [Document Revision History](#_grjogdjh5fi8)

| Version | Date | Author | Comments |
| --- | --- | --- | --- |
| 1.2 | <11/28/21> | Taylor Murphy | Made changes to Evaluation |

**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 client wants a game that one or more team are involved, and each team can have multiple players assigned to them. The game consists of four rounds lasting a minute each. If the team in play does not guess the puzzle before the time limit the other team is offered a chance to solve within a 15 second time limit.

## [Design Constraints](#_2et92p0)

* Multiple teams could be involved.
* Each game and Team names must be unique.
* Allow users to check if the name is already in use.
* Only One game can happen at one time.
* Web based and can operate on multiple platforms/

## [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 Entity opens a relationship between Game, Team and Player classes. They all get information from Entity. Each class will share references like id and name. We can observe their relationships and see that Team and Player is a “has a “ type. While Game has a Team and GameService has Games. We can also observe that Game Service has a reference of Games, Games a reference of Team and Team a reference of 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 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** | Pros: Mac has Work Group manager, allows for easy changes to user interfaces and and configure user environment initially where users can or can’t make changes. macOS server app allows easy and low-cost server hosting.  Cons: not often used for web hosting and possibly having to set user preferences and restrict user preferences based on individuals or groups. | Pros: is open-source operating system and is used by large portion of domains and supports public cloud servers, so support could be more accessible. It also the most popular webhosting platform. Also cost to use Linux is extremely low because it is free.  Cons: The expertise required for Linux is higher than other operating systems like having to work with command lines. Also Linux hosting is not compatible with windows apps. | Pros: Windows utilizes it own Server technologies Such as Windows server that is an application that allows for hosting. For the machines to run windows there is a large availability of different options and support.  Cons: Windows itself is not free and has a licensing fee of for download of 199.99 and windows server 2019 cost 499.99 for licensing and most desktop applications are not cross platform with Linux. | Pros. Operating systems such as android use a Linux kernel which is based on Linux with extended libraries. So this allows developers with knowledge of Linux to have an easier time to execute and use these systems.  Cons: Bandwidth limitations set to not allows the machine to be overloaded with request also limits overall usage as a server along with possible security concerns since it can be mobile and switch connections along with some high cost for licensing fees and application cost. |
| **Client Side** | Pros: Easy to use and understand user interfaces and “bare bones” style operating system.  Cons: High cost of machines, limited upgradability due to restrictive control from apple and possible limited preference settings. | Pros: it is easy to install and easy to run on most machines, so resource cost is low and is very resilient to viruses without requiring any additional software.  Cons: Linux doesn’t work well with other operating systems since it has proprietary apps and has high learning curve for users.  https://linuxnetmag.com/linux-pros-and-cons/ | Pros: The availability of windows machines for users are plentiful with many options and support along with around an 85% market share according to net market share. With this availability the level of knowledge to use is not high.  Cons: Many applications require licenses and a lot of background processes needed to operate the resource cost could be high depending or web browser being used by user such as chrome requiring a lot of RAM. Also could be easily exploited by viruses and requires additional software. | Pros: Unlike with server side since the user is allowed access to web-based applications along with browsers they have access to many options along with scalability of the fact that mobile devices are in high use. Plus, with this most mobile device are easy to use with easy-to-understand interfaces Such as IOS devices.  Cons: With this mobility come some security concerns since users can connect to any internet connection this can allow for viruses and unauthorized access. Also Android devices can have are more cumbersome user interface and cost of apps to run some of these application come at a cost. Lastly with the smaller devices the hardware limitations are rather strict and can become burdened very quickly. |
| **Development Tools** | Macs OS X has work group manager and macOS server app which is low cost 19.99 from apples website. Can use Atom which supports many different programming languages which also works with windows and Linux https://atom.io/ .  It also has access to many other development tools and apps that are cross platform but also many that are not cross platform. https://www.flexihub.com/best-mac-developer-tools/ | Linux has access to many open-source applications for programming languages and most of these applications support popular languages such as HTML, CSS and Java some these include Eclipse along with Atom.  https://www.pluralsight.com/blog/software-development/linux-open-source-development-tools | The development tools and applications available are plentiful along with atom and eclipse it also has visual studio which supports cross platform development along with mobile, azure and web development according to visual studios platform compatibility document. But it is also available on macOS and Linux which allows cross platform development. | There are many mobile app development tools that can support cross-platform development since mobile devices have such a high market share, and everyone uses them. Some of these can include React Native and unity3D. This availability of cross platform development also allows the use of many different and popular programming languages and libraries that are also used in other operating systems. |

## 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 Windows with its accessibility and easy to use build. Plus, with its access to a wide variety of software and hardware not available to another platform. Plus, it also a very popular OS amongst most users.
2. **Operating Systems Architectures**: Windows makes use of multiple services from other third-party companies and different applications use different webservices that can be used using a user account or server. Windows also supports many different applications such as Atom and has Visual Studio which is supported by other operating systems which will allow for easier cross development.
3. **Storage Management**: Storage management is easy to access on windows with easy-to-understand UI along with this you can see memory allocation that a software uses and you are able to change and choose locations to save software and applications.
4. **Memory Management**: This game could use a database for use of player information and will also use a library containing pictures. You can access memory allocation of an application and see what your IDE is doing and how much memory it is using.
5. **Distributed Systems and Networks**: Noting the differences between operating systems there are many ways to publish games on different platforms. Unity could be a good tool to utilize since it supports multiple operating systems and mobile systems. It can be free to use and is used by a lot of developers to create a lot of games and there is access to a lot of tutorials and information for unity along with unity.
6. **Security**: Security Software some of the time encrypts user data and send the encrypted files to the target machine which decrypts it with a key. Also, there are other implementations such as firewalls and the like. Although Windows come with prebuilt security and anti-virus software It would be recommended to invest other anti-virus software to support the amount of client traffic you might receive.

**Sources**

1. <https://www.computerworld.com/article/2564427/a-look-at-mac-os-x-server-s-client-management-abilities.html>
2. <https://digital.com/best-web-hosting/operating-systems/>
3. <https://digital.com/best-web-hosting/linux/>
4. <https://www.vpcart.com/blog/windows-vs-linux-hosting-the-advantages-and-disadvantages/#:~:text=Linux%20for%20servers%20is%20more,based%20applications%20on%20a%20server>.
5. <https://linuxnetmag.com/linux-pros-and-cons/>
6. <https://netmarketshare.com/operating-system-market-share.aspx?options=%7B%22filter%22%3A%7B%22%24and%22%3A%5B%7B%22deviceType%22%3A%7B%22%24in%22%3A%5B%22Desktop%2Flaptop%22%5D%7D%7D%5D%7D%2C%22dateLabel%22%3A%22Trend%22%2C%22attributes%22%3A%22share%22%2C%22group%22%3A%22platform%22%2C%22sort%22%3A%7B%22share%22%3A-1%7D%2C%22id%22%3A%22platformsDesktop%22%2C%22dateInterval%22%3A%22Monthly%22%2C%22dateStart%22%3A%222019-11%22%2C%22dateEnd%22%3A%222020-10%22%2C%22segments%22%3A%22-1000%22%7D>
7. <https://7t.co/blog/mobile-cloud-hosting-pros-and-cons-to-consider/>
8. <https://hackernoon.com/9-popular-cross-platform-tools-for-app-development-in-2019-53765004761b>