

<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 | 11/21/2021 | Roberto Crawford | Multiple teams of developers, an increased budget, and increased time. |

**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 wants to make a web-based game that can work on multiple platforms. The game they want to develop is called “Draw it or Lose it” and is currently only available for android. The goal of this game is to have multiple teams that contain up to seven individuals going about four rounds a minute. Once a picture is pulled from the library each time will have one minute to guess until times up. If the selected time does not answer correctly the opposing team will have an opportunity to answer for 15 seconds.

## [Design Constraints](#_2et92p0)

Some of the design constraints consist of needing multiple teams each containing an ample amount developers. The game name must be unique along with said team names to identify if it is in use. The game must also run-on multiple platforms. This will imply that the Android code may have to completely be re-written to accommodate the new platforms the game will be running on. Possibly incorporating different languages to make the code more robust.

## [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)

Entity creates a relationship between Player, Game and Team class. This implies all the classes gets information from Entity. Each class will share the same reference such as the “id” and “name”. The entity is more a less the superclass since every class communicates with it. The relationship between the game containing “teams” and the GameService containing “Games”. The Gameservice service contains references to games, team, and 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** | Some of the mac characteristics is A flexible terminal to configure the access, make some changes and configure a server if need be. Also, it is popular for web hosting. Some advantages are that has multiple web hosting options. Disadvantages consist of not being as popular.  This platform offers a server-based deployment where the website will be hosted.  Characteristics, advantages, and weaknesses are proprietary, secure, vastly more expensive, decent support, not that large of a user base.  Potential licensing cost will be minimal for the gaming room, none for the client, and a decent amount for the server operating system. | A very flexible terminal that can be configured for access, changes, and server adjustments if needed. It is mostly secure characteristics which make it most preferred. Some advantages are that Security flaws are assessed and caught before it becomes a possible issue. It is the most preferred OS for web hosting. The disadvantages is being more difficult to find applications to support it.  This platform offers a server-based deployment where the website will be hosted.  The potential cost to the clients is nothing since they can use a Linux based server, additional tools can be used at no additional cost.  Characteristics, advantages, and weaknesses are open source, secure, free, cost effective, less support, and a much smaller user base.  Will be no potential licensing cost for the server, the client, or the gaming room. Since Linux is an open source | Has more software options available when compared to the other OS. Characteristics are that it is the most used platform in comparison. Some advantages are the resource requirements are high, ease of use, and loading. The disadvantages are higher chance of getting a virus which is rare.  This platform offers a server-based deployment where the website will be hosted  Characteristics, advantages, and weaknesses are  Proprietary, more secure, expensive, more support, and vastly higher user base.  Will be no potential licensing cost for the client, will be a minimal amount for the gaming room, and a decent amount as well for the server. | Characteristics vastly More popular, affordable, and portable. Some advantages are having a wider reach amongst individuals and tons of compatibility. The disadvantages are selective to what accessories and applications are available for certain mobile devices, weak security.  This platform offers a server-based deployment where the website will be hosted  Characteristics, advantages, and weaknesses are proprietary, secure, various costs, good amount of support, huge user base.  No potential licensing cost for the client, will be a good amount for the gaming room, and a higher amount for the server compared to the other OS. |
| **Client Side** | Moderate knowledge about how a computer works. Cost very expensive compared to other platforms. Only can receive repair directly from Apple themselves.  For desktop application the development tool is XCode and the programming language is swift  A team that is familiar with this software will be needed to provide the best compatibility. | Highest amount of knowledge needed on how a computer works.  The cost can be similar to that of a windows but may be cheaper since it needs less resources to run.  For desktop application the development tool is Eclipse since it’s the most popular amongst Linux users, the programing language is c which is the base.  A team that is familiar with this software will be needed to provide the best compatibility. | Minimum amount of knowledge needed on how a computer works. Cost is moderately expensive or cheap depending on your requirements.  For desktop Application the development tool is visual studio, the programming language is visual basic which is the default. Can be quite difficult developing network-oriented applications with visual basic but it is still possible.  A team that is familiar with this software will be needed to provide the best compatibility. | Offers tons of flexibility having multiple OS for similar price points. Also provides Updates anywhere for the clients or developers. More difficult to access anything if not optimized for mobile services.  A team that is familiar with this software will be needed to provide the best compatibility. |
| **Development Tools** | Regarding to running languages on a mac, macs can run virtually anything. So well in fact its decently popular amongst programmers. Also, it contains useful application that may add in the programming process such as notepad. As well as supporting libraries for programs. Such as HTML/ JavaScript/java/python/C++/C and more  The impact will be minimal for the development team given how this software can support many languages given the right tools.  There is no licensing cost related to development tools. | Linux offers tons of flexibility given it can work with eclipse, Visual studio, and notepad. Giving an abundance of options for the user to choose from. This allows the user to code however they see fit. Linux allows you to run any language available, while supporting libraries for programs. Such as HTML/ JavaScript/java/python/C++/C and more  There would be little to no impact given how open-source Linux is giving way more flexibility in comparison.  There is no licensing cost related to development tools. | Windows offers the user a more seamless and easier experience in comparison to Linux. Windows can also run eclipse, visual studio and can also utilize notepad. Windows offers the user to run any program language available, while supporting libraries for such programs. Such as HTML/ JavaScript/java/python/C++/C and more  The impact will also be minimal since there is a vast number of individuals familiar with this OS and how it operates.    There is no licensing cost related to development tools. | Mobile devices are flexible to an extent but can still offer the user the ability to program. Tons of apps can be created on a mobile device, but more intensive application may require a computer to make. Some languages consist of HTML/ JavaScript/java/ python/ and more.  The impact will be stronger given the vast number of devices with different configurations may pose some difficulty if not familiar with it.  There is no licensing cost related to development tools. |

**Client side**

**Desktop apps can a great choice offering tons of flexibility given the user a pleasant experience in the process. During the application development process optimizing and multiple teams that are familiar with a specific OS to allow the most stable experience for the users.**

## 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 Operating platform I recommend would be windows since it is vastly more popular and readily available to the most users. Windows does not require a massive budget and minimum expertise of the OS to start projects. Also, windows offer an abundance of tools readily available.
2. **Operating Systems Architectures**: Windows provide services that Is used by applications based around windows that allows applications to show a graphical user interface, Including accessing system resources. These application services can be used through an account or a server.
3. **Storage Management**: Windows offers a viable feature regarding storage called “storage sense”. This allows the user to manage all their files and identifying how much space it occupies. Another feature that is present is having the option to choose where you want for your apps to be stored making the process much smoother. Also, can use the Cloud in case a problem a problem where to occur or the file is too large to be contained on a hard drive or SSD. The cloud would be recommended given the circumstances of the game room wanting to add more content in the future.
4. **Memory Management**: Since this game has a vast number of pictures it may need to be stored along with various other things. Memory allocations offers an easy way to store pictures into a separate file besides the default one. This allows you to keep mostly your whole project in the same file rather than going on a scavenger hunt. This keeps opening files to a minimum when working with other programs.
5. **Distributed Systems and Networks**: Since each operating system is not the same I found a possible tool that may assist in enabling cross platform amongst different devices. React Native seems to be the highest rated IDE when it comes to enabling cross platform. Apparently once the game is created you can export the file to the internet, Android and IOS which will allow cross play. To prevent possible outages or connection issues the company will need to insure powerful enough servers to support the demands of a large volume of players Including backup power for said outages.
6. **Security**: Windows already comes with built-in protection software called “Windows Security” but to secure the privacy of the user another source would be recommended. The Windows protection is more than enough regarding the system it can scan for malware, security compromises, and viruses. This protection service regularly updates in case of new threats.