

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 | 7/14/2022 | Steven Colley | Web app development of an android based app. |

**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 has an android-based game in which players have 30 seconds to guess what is being drawn. If the team doesn’t guess the image than other teams have 15 seconds to steal. There are multiple teams of several people that participate through 4 rounds. The Gaming Room would like their android-based game to be developed as a web application that can be ran on multiple platforms.

## [Design Constraints](#_2et92p0)

The game is already programed for android but need to be developed as a web application. This means that the coding will need some assistance to run on multiple web-based devices such as desktops, smartphones, and tablets. The game also requires a lot of people to commence. There are 4 teams in total consisting of multiple players. This may make things difficult when it comes to testing. The game and team names must be unique to allow user to check whether name is in use or free. Only one instance of the game can exist at any time.

## [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 has a relationship of parent class for three child classes: Player, Team, and Game. There can be multiple instances of the three child classes, but Game can exist without Team and Team can’t exist without Player. GameService also can’t exist without the Game class being instantiated. For the Game state the program will use a singletonTester to create only one instance of the game.

**"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** | Mac as an operating system is easy to use however their operating system isn’t designed to be used as servers. Mac did have a service called macOS server but as of the year 2022 that service will no longer be in use. | Linux is known for its open source that provides a greater community support. Linux also isn’t as used by many users, so security threats are low. It’s also free to use. | Windows is easy to use with great third-party support but has paid licensing. Windows also is at a higher risk of security threats due to its high user base. | Although can be done and once supported by nokia mobile web servers aren’t intended for large companies with web applications. Mobile servers work best with HTTP sites. |
| **Client Side** | Cost is high with mac as brand only develops high-cost products. Low expertise is required. Low time is required. | Low cost with Linux OS. High expertise is required. High time required. | Medium cost is required for Windows OS. Medium expertise. Medium time required. | Cost medium to high for client. The expertise required is low for users due to the wide adoption of users. Low time is required by users. |
| **Development Tools** | Developmental languages for web-based applications is traditionally consist of JavaScript, CSS, and HTML for frontend. IDE’s consist of JavaScript, Python, and PHP. The development tools consist of PyCharm, Eclipse, VS, and Notepad++. | Developmental languages for web-based applications is traditionally consist of JavaScript, CSS, and HTML for frontend. IDE’s consist of JavaScript, Python, and PHP. The development tools consist of PyCharm, Eclipse, VS, and Notepad++. | Developmental languages for web-based applications is traditionally consist of JavaScript, CSS, and HTML for frontend. IDE’s consist of JavaScript, Python, and PHP. The development tools consist of PyCharm, Eclipse, VS, and Notepad++. | Although can be done tools aren’t as widely available for developing on smartphones. Android is the only system of mobile that is possible but file management isn’t as good as other options and IDEs aren’t as intuitive. |

## 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 that the Gaming Room uses the windows platform. Windows although more expensive than the linux operating system has less of a learning curve. It is also cheaper than operating on mac devices.
2. **Operating Systems Architectures**: The architecture of windows includes a user mode which is restricted to what is allowed for the user and a kernel mode which is unrestricted.
3. **Storage Management**: using a storage management provider would allow the storage to be configured is a multitude of ways which would include using of a SSD, HHD, and external storages.
4. **Memory Management**: Windows allows for a wide range of memory management options which is dependent of the version of windows being used. The memory for a Win32 app can range from 2GB up to 2TB. Windows also allows for the allocation of memory which uses would limit the amount of memory being used.
5. **Distributed Systems and Networks**: Looking into way for development for multiple platforms I came across libGDX. This would work for our development because it’s a java based development framework that allows developers to create for windows, linux, macos, android, and ios. Having a single framework would allow for cheaper development of the game and help reach our goal of usability across multiple platforms.
6. **Security**: For security we could use the built in windows defender which already scans for malware, viruses and other threats.