

<The Gaming Room>

# **CS 230 Project Software Design Template**

Version 1.2

## 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 |
| --- | --- | --- | --- |
| 1.2 | Aug/21/2022 | Hamza Malik | Evaluated with different OS, Adding Recommendation section |

**Instructions**

Page one (the cover page) should include all the information listed in brackets, as well as in the table of document revision history and under each heading. Delete the bracketed prompts under each title and replace them with your own paragraphed answer addressing the facts presented.

## [Executive Summary](#_sbfa50wo7nsh)

Gaming Room seeks to create a web-based game that can operate on a variety of devices, such as smartphones and tablets. "Draw It or Lose It" is the name of the game, which is presently exclusively accessible on Android. The goal of this game is for teams of numerous individuals to compete in four rounds of one minute each, with each team consisting of four persons. It's a guessing game where a picture is drawn from a library of photos and the timer runs out. A member of the opposing side is given 15 seconds to respond if the question is not answered.

## [Design Constraints](#_2et92p0)

* Needs one or more teams involved
* Each team has multiple people
* Game and Team names must be unique to allow users to check whether the name is in use or free
* Only one instance of the game can exist at any time.
* Must run on multiple platforms

Requirements for producing code and software that must be adhered to Aside from game creation, we still need to look at application development. All devices in the Gaming Room are required to be able to run this. Android is already installed; however we need to get it working on a different mobile platform. Windows, Linux, and Apple computers are just some of the options available in this category. In order to achieve this, we will need to either re-write the code in Swift for Apple devices or come up with a mechanism to leverage existing code to operate on other devices by inheriting the language of the original. Like when we combine several programming languages to create better code.

## [System Architecture View](#_ilbxbyevv6b6)

Please note for these projects, there is no need for this part, but it serves as a reminder that specifying the application's system and subsystem architecture, including physical components or tiers, may be necessary for future projects. Communication and storage network topology should be presented in order to fully comprehend the design.

## [Domain Model](#_8h2ehzxfam4o)

Relationship between the Game, Team and Player class is established by the Entity. This implies that Entity serves as a source of knowledge for all of them. This may be shown using inheritance in a UML diagram. As a result, each class will have a common "name" and "id" to refer to. Creating a superclass for Entity. Team and Player have a "has a" sort of connection, which we can tell by looking at their relationship. In contrast, both the Game and the Game Service have a Team. We refer to it as aggregation while working with UML (HAS-A). An instance of one class is linked to an instance of another class when a user possesses "a." This diagram shows that Game Service has a reference to Games, Games to Tea, and Teams to 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)

Consider the following needs and provide your conclusions for each of the operating systems (Linux, Mac, and Windows), as well as mobile devices, based on your experience. Make sure to keep your client's needs in mind as you finish the table, since it all has to come together.

Your answer should be a whole paragraph that includes all of the relevant information in each cell.

| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| --- | --- | --- | --- | --- |
| **Server Side** | Flexible terminal commands for configuring the server or making modifications are available.  Characteristics It is widely used in web hosting services.  Advantages It's expandable, with a wide range of choices to meet a wide range of web hosting needs.  Disadvantages It isn't as popular as it once was among web hosting companies. | Additionally, Macs are more cost-effective than PCs.  Characteristics Secured is the best option.  Advantages Security problems are spotted before they become a problem, making it the most popular option for web hosting services  Disadvantages Web hosting-related apps are harder to locate. | Compared to other operating systems, there is a greater selection of applications.  Characteristics It has a suffocating hold on the competition. Platforms should be closed.  Advantages High demand for resources, reduced loading time, and a high level of comfort  Disadvantages viruses are easily spread, and tech help is mediocre. | Having a server that stays put and can be monitored from a single location is preferable. Other devices have superior specs.  Characteristics Popularity and portability make it a popular choice.  Advantages A bigger audience, improved compatibility, and a more affordable price point  Disadvantages A wide range of smart mobile devices are not compatible with it. Inadequate protection |
| **Client Side** | Requires a fair amount of knowledge and effort on your part. Windows have a comparable price tag. All web browser platforms and mobile devices must be supported by an app's development process in order for it to function properly. | Requires much knowledge and time. Affordability at the barest minimum. All web browser platforms and mobile devices must be supported by an app's development process in order for it to function properly. | Requires the least amount of knowledge and effort. The price is comparable to that of a Mac. All web browser platforms and mobile devices must be supported by an app's development process in order for it to function properly. | Allows customers and even developers to access changes at any time and from any location. The implementation of this gadget is a little more complicated than with others. |
| **Development Tools** | Swift is the most often used language runtime on Macs. Notepad++ and other useful apps may also be thrown in. Even so, Macs are able to run any language. The front-end and general-purpose languages are supported by libraries that include, but are not limited to, HTML/CSS/JavaScript. Some examples of this are Ruby on Rails, Java, Python, and PHP. | It's possible to use Linux with visual studio, eclipse and notepad++ for an easy-to-use development environment In addition to a wide range of languages and applications. The front-end and general-purpose languages are supported by libraries that include, but are not limited to, HTML/CSS/JavaScript. Some examples of this are Ruby on Rails, Java, Python, and PHP. | Easy to use yet just as powerful as Linux. So, for example, among the various programming languages, there is Visual Studio and Eclipse. Notepad++ is an easy-to-use programmed because of its many features. The front-end and general-purpose languages are supported by libraries that include, but are not limited to, HTML/CSS/JavaScript. Some examples of this are Ruby on Rails, Java, Python, and PHP. | Android and fast let you to construct an almost limitless number of applications. All three machines can run the same languages and applications. Front-end languages include but are not limited to HTML/CSS/JavaScript, whereas general-purpose languages are supported through libraries. Java, Python, PHP, and Ruby are just a few examples. |

## Recommendations

Analyze the characteristics of and techniques specific to various systems architectures and make a recommendation to The Gaming Room. Specifically, address the following:

**Operating Platform** When comparing mobile games to PC games, it's important to keep in mind that the two platforms are not interchangeable. We are well aware that each console, PC, and mobile game has its unique set of pros and disadvantages. To ensure the longevity of their own platforms, all platforms fight for the highest user rating. Each gaming platform, in an effort to improve their rating, produces games of a higher quality. This means that game makers for each platform need to put up their best effort to remain competitive, as the quality of their respective platforms will be determined by the games available for them. So, Windows is strongly suggested.

1. **Operating Systems Architectures**: For all Windows-based programmers, Windows offers services such as a Graphical User Interface (GUI) and access to system resources that allow apps to exhibit a GUI and much more. Visual and Multimedia applications as well as Messaging and Web Services are all included in this category. User accounts or servers may be used to access these services.
2. **Storage Management**: Storage sense is a fantastic addition to Windows 10. Using this feature, you may examine and manage the files on your hard disc, as well as the amount of space they take up. Other advantages include the ability to store app locations, which makes it simpler to locate them. For data storage, you may also utilize the cloud, as with other things. For huge projects, the built-in storage system makes it simple to create and put files, ensuring that they won't be lost or accidentally destroyed.
3. **Memory Management**: You'll need a lot of images to populate your database or library while working on this project. Allows for convenient storing of photographs outside of the normal picture folder thanks to the RAM allocation, this enables you to save your entire project files in one place on your computer, making it safer. It's important to remember this while you're using your IDE to build the game and accessing files inside it.
4. **Distributed Systems and Networks**: In order to ensure that the game will operate on every operating system, I looked into options to publish the game. I discovered Develop 4, a cross-platform game development tool. Any device may be used to execute it since it is an IDE. As soon as you've finished making the game, you can easily export it to platforms such as the web, iOS, Android, and more. With this, it will be easier to manage a network's interdependencies. As a result, the corporation must ensure that its servers are powerful enough to sustain big player volumes and have backup power in the case of a power loss.
5. **Security**: The security software that comes preinstalled on Windows is rather robust. However, using a different source would be preferable for the sake of data security for users. However, if we're talking about the software installed on the computer, Windows already has security features built in. Use this tool to check your computer for viruses, spyware, and other forms of dangerous software. To keep the system and user information secure, this all occurs in real-time and is automatically updated as threats evolve.