

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 | 07/17/2022 | Tristin Watson | Summary, Design Constraints, and UML Description |

## [Executive Summary](#_sbfa50wo7nsh)

The game needs to have one or more teams and each team can have multiple players. Team names must be unique. Lastly, the game can only have one instance at a time. The game uses four rounds of guessing a picture taken from a stock library. The team at bat has 30 seconds to guess the picture or the other teams get a chance for 15 seconds afterwards. The game needs to be web-based.

## [Design Constraints](#_2et92p0)

Game must be web-based. The game needs to be singleton, so that only one instance is allowed. The interaction must be smooth and easy to follow. There must be multiple teams, but unique team names can be chosen.

## [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 program driver method uses a singleton tester method. GameService class has a games list, nextGameId, nextPlayerId, nextTeamId, and service private methods and getters to use those methods. The GameService class also has a constructor and a singleton method. The Game class has a private teams list, and a few public methods called addTeam and toString. A constructor is also available in the Game class. The team class has a private players list, and two public methods addPlayer and toString. Team class also has a team constructor. The player class possesses one public class called toString and a constructor. The last class in called Entity. Entity has two private variables called id and name. There is also a constructor for the Entity class along with two getter methods and a toString method.

**"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)

| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| --- | --- | --- | --- | --- |
| **Server Side** | Advantage: Easily accessible interface. Configures and manages built in services with ease.  Virus attacks are infrequent.  Weakness:  High Cost.  Hardware is customizable.  Mac file systems cannot be read by Windows.  Characteristics:  Open directory.  Time machine back up service. | Advantage:  Open source.  Free installation.  Weakness:  Software compatibility.  Hard to understand unless adept at computer usage.  Characteristics:  Portable Environment.  Free and open source.  The infrastructure is lightweight. | Advantage: Excellent security. Accessible software.  Weakness:  High cost.  Single point failure.  Characteristics:  Windows Admin Center.  Client Connections are automated. | Advantage:  Cloud data backup.  Cheap and scalable.  Weakness:  Low security through cloud data.  Limited bandwidth to prevent over server burdening.  Characteristic:  Cloud storage backup data.  Fast speed because the technology is increasingly modernized. |
| **Client Side** | Advantage:  Optimized software integration.  Clean Interface.  Weakness:  Restricted App Availability.  Expensive development costs. | Advantage:  Open source.  Efficient resource use.  Easy updates.  Weakness:  Limited market share.  Hard technical curve.  No hardware support. | Advantage:  High performance.  Reusable components.  Weakness:  Caching problems.  High browser memory usage if cached wrong. | Advantage:  Ease of access.  Data gathering is quick.  Weakness:  Regular updates required.  High cost for app development.  Complicated design process. |
| **Development Tools** | Main used programming languages for Mac are JavaScript and SQL. SQL is used for the server. | Most common languages used for Linux servers are Python and C++. | Windows server coding languages that are used the most are C++ and C#. Closely followed by JavaScript. | Programming languages uses most often for mobile devices are JavaScript, Kotlin, and C++. |

## Recommendations

1.       **Operating Platform**: The operating platform best suited for The Gaming Room would be the operating system used for mobile devices.

2.       **Operating Systems Architectures**: Mobile applications have a few wonderful advantages such as they have a Cloud data backup, and they are cheap and scalable. Mobile devices are modern technology, so they naturally have high-speed processing power. Mobile devices are also mobiles as it says in the name. Meaning that anywhere you wish to play the gaming application, you can do so to your heart's content.

3.       **Storage Management**: Mobile devices can store most data on the cloud. The cloud storage can easily be backed up when necessary, so it is unlikely that data will be lost. You can also store data that is not in immediate use on the mobile device itself to be recalled whenever it is needed in the use of the application. Since storage is the data that is needed for running that application but not needed for immediate operations, you can store the data however is needed to get the job done. It is recommended, however, to not pull a Titan Fall and have 40GB of uncompressed audio files for your application download. Storage is long-term data that is not needed during most functions of the application but is still data needed later to run the app optimally.

4.       **Memory Management**: Memory management for mobile devices is also as simple as storage management. The difference is that memory is what is needed for immediate processes during the running of the application. The best way to describe it would be the data that is processed continuously to keep the application running at tip-top performance. Memory is short-term data that is not needed for long-term functions.

5.       **Distributed Systems and Networks**: The best way to keep an application up and running on various platforms would be to make sure you always have a secure internet connection. Ethernet cables are a must-have for any server hosting. Ethernet cables provide a direct connection to the internet provider. While wireless connection for servers is possible, it is highly likely that a wireless connection can get mired in functions and slow down. A wired connection, however, might not be possible for a mobile device server. If that is the case, it is best to reduce functions that must be accessed on the server. Keep wireless server functions limited and most functions client side except for vital information and updates.

6.       **Security**: Security is a must-have for any application that is being run and operated. Some useful security measures are auto clean, automatic updates, real-time scanning, activity auditing, and limits for user access. Auto clean helps remove viruses from your software. Automatic updates make it to where you do not need to manually update out-of-date software. Real-time scanning can prevent those with wicked intentions from accessing the data you wish to protect. Activity auditing allows you to view user activity, such as log-in, log-out, and what information was accessed or attempted to be accessed. Lastly, limited user access makes it where users cannot just access critical information on a regular user account. You can set certain permissions to certain accounts such as an Admin account. Although the list may seem long, these are all incredibly useful security features that create a balanced and secure application. You can protect yourself and your users properly with these security measures.

**Resources**

Krause, J. (2021, September 1). *Top 7 features of windows server 2019: Worksighted Blog*. Worksighted. Retrieved July 31, 2022, from <https://www.worksighted.com/infrastructure/windows-server-2019/>

Nash, A. (n.d.). 7 advantages and disadvantages of Mobile Device Management [2022]. 7 Advantages and Disadvantages of Mobile Device Management [2022]. Retrieved July 31, 2022, from <https://mobiletrans.wondershare.com/mobile-device-management/advantages-and-disadvantages-of-mdm.html>

Nonis, S. (2022, March 15). Linux features: 10 best features of linux operating system. MonoVM.com. Retrieved July 31, 2022, from <https://monovm.com/blog/Linux-Features/>

CityMac. (n.d.). Overview of OS X server. citymac.com. Retrieved July 31, 2022, from <https://www.citymac.com/blog/2016/04/04/overview-of-os-x-server>

Long, S. (2019, September 27). Mobile cloud hosting: Pros and cons to consider. 7T, Inc. - SevenTablets. Retrieved July 31, 2022, from <https://7t.co/blog/mobile-cloud-hosting-pros-and-cons-to-consider/>

Mews, J. (2021, September 28). The advantages and disadvantages of a mobile app 2021 2021. Objective. Retrieved July 31, 2022, from <https://objectiveit.com/blog/the-advantages-and-disadvantages-of-a-mobile-app/>

Supersu. (2021, March 21). Top 10 advantages and disadvantages of linux. KernelTips. Retrieved July 31, 2022, from <https://kerneltips.com/top-10-advantages-and-disadvantages-of-linux>

Lewis, D. (2022, February 15). MacOS review: The pros and cons. Konsyse. Retrieved July 31, 2022, from <https://www.konsyse.com/articles/macos-review-the-pros-and-cons/>

Singhal, G. (2019, December 19). Gaurav Singhal. Pluralsight. Retrieved July 31, 2022, from <https://www.pluralsight.com/guides/pros-and-cons-of-client-side-rendering>

Spanbauer, S. (2008, November 25). How to set up a cross-platform network. Entrepreneur. Retrieved August 13, 2022, from <https://www.entrepreneur.com/article/198868>