

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 | 11/10/22 | Joseph Daniszewski | Added Summery, constraints, evaluation, and recommendation. |

**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 develop a web-based game that can serve multiple platforms based on their current Android app game. The game is called “Draw it or Lose it”, a puzzle game. The game consists of four rounds of 1 minute. Teams need to guess the puzzle within the time limit, or the opposing team can guess. Drawings are rendered at a steady rate and fully conceived by the 30 second mark.

## [Design Constraints](#_2et92p0)

* Able to run on multiple platforms.
* Only one instance of the game can exist in memory at any given time. More than one team is required for each game.
* A game will have the ability to have one or more teams involved.
* Each team will have multiple players assigned to it.
* Game and team names must be unique to allow users to check whether a name is in use when choosing a team name.

## [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)

Game, Team, and Player classes will all inherit from the Entity Class, giving those classes access to the ID and name attributes and the getName and getiD methods. The gameService, Game, Team and Player classes are all associated with each other and referenced with a “zero to many” aggregation starting with the GameService class and working right to the Player class. The Entity class is using a private constructor to restrict multiple instantiations.

**"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** | More expensive than other platforms.  High stability, less prone to bugs and crashes.  Updates are regular and fix issues quite fast. | Open source and considered more secure than Windows.  Poor software compatibility.  High processing power. | Almost any software, no matter how obscure will almost certainly run. High software compatibility.  Very bug prone, lower security. | +One of the most common devices used today.  -Development is more difficult and will require more time to port over.  -High variety between devices due to manufacturer limitations or |
| **Client Side** | Mac is the most expensive of all the listed options. Macs tend to be priced higher than similar models on other platforms. Although Apple does a great job of keeping their machines updated and running for a long time.  As most people grew up using windows systems, Mac does have a learning curve and require developers familiar with it. | The most obscure of the big 3 (less than 1% of PC users use Linux. Therefore, competency with this operating system will have the smallest subset of individuals.  Can purpose build a PC to keep cost hardware costs down.  Free. | Low-cost operating system. Mass familiarity with the system, all developers are guaranteed to have knowledge so will not require much time of expenses to start up.  Can purpose build a PC to keep cost hardware costs down. | Nearly all development constraints will be higher using mobile devices to develop.  Higher costs, lower security, more time and less expertise, translating to more training.  I do not recommend this method. |
| **Development Tools** | Mac is streamlined for using Xcode IDE, you can also use it for Java although it is not recommended.  Most popular IDE’s you can get online are designed to be cross platform and can run on Mac.  C languages are more difficult on Mac as they lack many of the directories Windows has and uses. Making it more time consuming to develop working code. | Most modern IDE’s will run without issue on Linux. And due to the open source nature, they can run quite quick.  Can create a windows environment for development using virtualbot of the likes. | C languages and Java are much easier to develop in on this platform. Wide selection of native windows directories to use, making development faster and less complicated.  Most popular IDE’s you can get online are designed to be cross platform. Wide selection to choose from.  Can run Linux tools. | Very poor in this aspect. It is not a user friendly way to develop. |

## Recommendations

1. **Operating Platform**: My recommendation would be to use a Linux operating system. This is due to being highly secure, more affordable, modifiable to the needs of the business, streamlined, and having good capabilities of storage and memory management. Additionally, because of the widespread use of Linux, server administrators are more experienced and can provide better support for Linux-based machines.
2. **Operating Systems Architectures**: The Linux architecture is largely composed of elements such as the Kernel, System Library, System, and Shell functions. The Kernel is responsible for each of the primary duties of the Linux OS. System Libraries are a set of library functions implemented by the operating system and do not require code access rights on the kernel modules. System Utility Programs perform specific and individual jobs. Different operating systems are classified as graphical shells and command-line shells.
3. **Storage Management**: One storage management system that can be used with the Linux platform is FS-Cache. FS-Cache is a caching system that can be used to cache files from remote file systems, such as those accessed through NFS. It works by caching data from remote file systems to free space on the disk and maintains a certain amount of free space by discarding old objects from the cache. It also allows for the encryption of file systems at mount using eCryptfs, providing an encryption layer on top of an actual file system. SQLite would be another good option. It works well with Javascript, which being a web app will probably use a substantial amount of it. It also doesn’t require configuration to start. Can be moved to a server or cloud location.
4. **Memory Management**: The Linux platform uses a variety of memory management techniques to support web applications. These techniques include virtual memory, page cache, anonymous memory, compaction, reclaim, and huge pages. Virtual memory allows the platform to separate physical memory details from the application software, allowing only necessary details to remain in physical memory. Page cache is used to store data into memory from hard disk. Anonymous memory is used for heap and stack within the program, and compaction is used to partition the pages with virtual memory. Reclaim is used to free up memory pages that are no longer needed, and huge pages are used to optimize the usage of scarce resources.
5. **Distributed Systems and Networks**: The distributed systems must have a network that connects all components (machines, hardware, or software) together so they can transfer messages and communicate effectively. Middleware can be used to simplify the connectivity between applications, application components and the network that connects the devices. The network should be reliable, with minimal outages, and be able to handle the data load of the application at any given time, this is especially important during launch when the initial users will be higher than normal.
6. **Security**: Protecting user information on and between various platforms requires an understanding of security measures that can be implemented. On Linux platforms, SSH keys, firewalls, VPC networks, and VPNs can be used to enhance security. SSH keys provide authentication and access control by creating a private and public key pair. Firewalls can ensure that access to software is restricted according to the categories of public services, private services, and internal services. VPC networks provide a more secure connection among resources by making network interfaces inaccessible from the public internet. VPNs can be used to create secure connections between remote computers and present the connection as if it were a local private network. These measures can be used to secure user information and protect it from unauthorized access.