

Draw it or Lose it

# **CS 230 Project Software Design Template**

Version 2.0

## 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 |
| --- | --- | --- | --- |
| 2.0 | 8/9/2021 | Jonathan Courington | New executive summary, design constraints and domain model |

**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 clients wish’s us to build a multiplatform, web distributed version of their game ‘Draw it or Lose it’. Java will be the language of choice and the platform will be built in it.

## [Design Constraints](#_2et92p0)

We will use a singleton model for the processes and use the Java API to complete it. It must be available on other platforms for this to be a success.

## [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 domain model uses various forms of object oriented programming most notably inheritance. You will notice below that the classes Game, Team and Player all inherit from the Entity Class giving them cross functionality with it, further the Game class inherits from the GameService class giving it extensive usefulness in the endeavor. Finally the singleton tester class will use the program driver to complete it’s tasks.

****

## [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.

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**: There are many differences between mobile games and pc games. The differences include the game, the play and the user interaction. All the platforms are in competition for this business and each one has its own characteristics. The more you develop your games the more you will understand what platform you need.
2. **Operating Systems Architectures**:

ARM: arm is for mobile gaming and not as powerful as X86

X86: This platform has a lot of support and is very powerful.

1. **Storage Management**: There are several different types of storage available, HDD is considered to be a slow alternative to SSD which is considered fast. There are different architectures associated with them in particular SATA and NVME
2. **Memory Management**: A 64 bit windows machine has a virtual address that is approximately 8gb in size for an application. The threads associated with each address can only be used for a single application.
3. **Distributed Systems and Networks**: A networked game typically has a database that is shared by all the players each of which is implemented from scratch. It also contains a paper or artery that is designed to keep a high application program interface.
4. **Security**: There is always a risk of data be communicated to an unwanted person. Data loss or the loss of access are threats as well, you could suffer serious financial loss. Some websites use anonymixation to reduces risk for there clients.