



Draw It or Lose It
CS 230 Project Software Design Template
Version 1.0

Table of Contents

CS 230 Project Software Design Template	1
Table of Contents	2
Document Revision History	2
Executive Summary	3
Requirements	3
Design Constraints	3
System Architecture View	3
Domain Model	3
Evaluation	4
Recommendations	4

Document Revision History

Version	Date	Author	Comments
1.0	09/10/2023	Eduardo Rodrigues	Domain Model
2.0	10/15/2023	Eduardo Rodrigues	Deployment Recommendations

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

The development team at, "The Gaming Room," request consultation on the prototype to be used in the formation of a real-time multiplayer game for their game, titled, "Draw It or Loose It". Our goal will be to shorten the time to deployment and give the client a plethora of hosting and platform options backed by support they can rely on with Unities Game development platform.

Requirements

The prototype will include a single service that draws from three types of resource containers to scale and manage a system of game generation that features a multitude of configurable and uniquely identifiable team and player options.

Design Constraints

The demo code in Java will be translated to the C# that is ubiquitous throughout Unity. The platform gives the client access to the various assets, networking, industry specific tools and support to get a game up and running for multiple platforms dependent on the client's choice of hosting and/or publishing option(s).

System Architecture View

Use of Unity will help us flesh out the remaining interfaces of our system, they even have completely automated deployment pipelines and services in place depending on the client's specifications. Our refactored code can be easily adapted to include new features required by a growing system.

Domain Model

The diagram below shows us our service, GameService , that can be uniquely instantiated by the select entry point, by ProgramDriver or SingletonTester. GameService can then be scaled with various configurations of Game, Team, and Player entities – where GameService houses Game instances and each subsequent entity houses the next entity over, Player being the lowest denomination of the configurables. Having the base class entity here lets us easily enhance the GameService resources to meet any sort of requirements that are needed with the addition of new interfaces.



Evaluation

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 must 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	Unity offers several multiplatform hosting options. The hardware in play will have a server that needs to be configured to be able to deliver the game to various browser versions. Unity is royalty free, but charges for by the seat for professional projects. Unreal uses a royalties-based compensation and may have pricing for their hosting options that are comparable to those used by Unity.			
Client Side	Given that the game mechanics lend themselves to using a cursor and keyboard – developing for the browser is a good option to reach a wider range of users, although unity does offer several multiplatform options.			
Development Tools	Unity hub is available for Windows, Mac, and Linux. There is also a complete development and hosting pipeline in place with various publishing options. A tool like Maven can be used to orchestrate the entire project management and development pipeline and more.			

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:** Unity offers a complete development and deployment package. Unreal Engine is equivocal and bolsters a higher caliber graphics engine. We can easily translate the Android game to be able to run as an HTML5 game using Unities API.
2. **Operating Systems Architectures:** Unity and Unreal can target multiple platforms. They also have a pipeline in place that will facilitate the design of an application that is usable in the client-side browser. Developers and the like have options to develop on their platform of choice. A deployment on a Linux virtual instance will bolster the most efficiency because this is not an enterprise application. This will not only cost less but scale and provide more flexibility with development, deployment, and maintenance.
3. **Storage Management:** Both companies cater to industry specific data warehousing needs and requirements for AAA Game titles, the film and music industry, and in Graphics simulations. This will be necessary for storage of the images used for teams to interpret. This is all dependent on the remote hosting option and how the deployment is planned out.
4. **Memory Management:** The DevOps platforms in play from either platform will allow the client to test and scale the system to be able to meet the projected load balance on the server. Unity provides various tooling that will help facilitate modeling runtime analysis and simulations of deployments during alpha and beta testing and beyond.
5. **Distributed Systems and Networks:** Using WebGL and HTML5 based multiplayer browser game lends itself to the use of the client/server model where various channels open between client browsers and the server holding the game. The vendors may be able to provide the telemetry and marketing data needed to distribute, monitor, and adapt the schema to optimize the user base. Emerging techniques at the edge can also be considered for efficiency and time to market.
6. **Security:** Both vendors have security options outlined for various deployment types and publishing needs. This is especially important for the development pipeline and within the networking library used for multiplayer assets. Hosting options will provide specific networking security features like external firewall and virtual machine hardening guides.