



Draw it or lose it
CS 230 Project Three
Version 1.2

Table of Contents

CS 230 Project One	1
<u>Table of Contents</u>	2
<u>Document Revision History</u>	2
<u>Executive Summary</u>	3
<u>Design Constraints</u>	3
<u>Domain Model</u>	3
<u>Development Requirements</u>	4
<u>Recommendations</u>	6

Document Revision History

Version	Date	Author	Comments
1.0	07/13/2022	Adam Resemius	Created the document and added the table of contents, document revision history, executive summary, design constraints, and domain model.
1.1	08/02/2022	Adam Resemius	Created the Development Requirement Chart for Server Side, Client Side, and identified Development Tools.
1.2	08/17/2022	Adam Resemius	Created the Recommendations page consisting of what operating platform, operating system architectures, storage and memory management, Distributed Systems and Networks, and Security.

Executive Summary

- Company Name: Game Room
- Type: Gaming/ Entertainment
- Goal: The Game room wants to expand their android application “Draw it or Lose it” to a web-based game. The client wants the game to run on multiple platforms. They expect the web-based version of their game to follow a set of software requirements. The game consists of multiple teams that have several people, they play four rounds, each one lasts a minute. A picture is displayed from a library full of images. A team guesses until timer ends. When the timer ends and the photo isn’t answered, each opposing team member gets to answer until a 15 second timer ends.

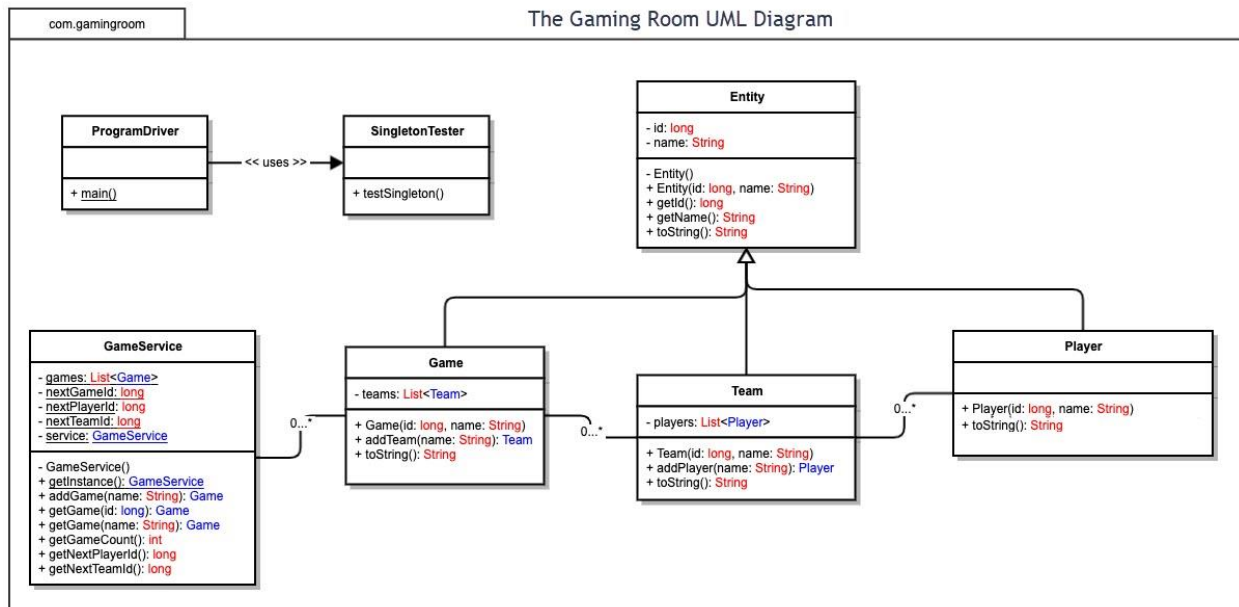
Design Constraints

- The game needs to have one or more teams playing
- Each player will be set into teams with multiple team members
- The game and team names need to be unique to allow players to check
- Only one instance of the game can exist in memory at any given time.
- The web app must be able to run on different hardware components
- Must have a singleton to create unique identifiers
- There needs to be an entity class that will inherit from the other classes.

Domain Model

The Class entity inherits all the public attributes for the Game, Team and Player classes. This means that the entity class has the functions and variables with the “+” next to them. The entity class is able to use the functions and have these variables of its own. Encapsulation is shown between the “+”, and “-” characters. If the character is “-” this means that it is private. Setting a variable to private such as Teams in the Game class, allows nothing but attributes in the

game class to manipulate that variable. This is called encapsulation. This is the case with all private attributes. Abstraction is then shown by hiding attributes like Game service from being seen from users. Entity is a super class because it inherits from multiple classes. This diagram also shows that game Service references Games, Games references Team, and team references Player.



	Mac	Linux	Windows	Mobile Devices
Server Side	<p>Creates flexible terminal commands to that allows the system to access or make changes by configuring the OS.</p> <p>Characteristics: Popular platform for hosting.</p> <p>Advantages Allows for easy updating/upgrading Easy to configure for hosting.</p> <p>Disadvantages Less used in hosting in the community</p>	<p>Uses terminal commands to access and change by configuring the OS.</p> <p>Characteristics Very secured, has the most likes</p> <p>Advantages Most preferred OS to browse the web, Problems are typically fixed before issues settle in.</p> <p>Disadvantages Difficult to find applications. Stricter due to security features.</p>	<p>Allows more third-party software compared to other OS</p> <p>Characteristics Dominant and highly used by most PC users.</p> <p>Advantages Allows less loading time, creates a great environment, updates quick Uses less storage due to better storage managements</p> <p>Disadvantage Viruses are common to run into, not enough support within the network.</p>	<p>The app can be used at any time and used for the best option due to that.</p> <p>Characteristics Very portable Can be used at any time of the day.</p> <p>Advantages Cost of development Very easy to start and use</p> <p>Disadvantages Multiple devices to code across Poor security Can be launched across multiple IOS and Andriod devices.</p>
Client-Side	<p>Apps are easy for the developer to update and make it available to the everyday user daily.</p>	<p>Takes time and money to create, as the development cycle is long and tedious with more security features.</p>	<p>Costs are similar to mac. The development is simple and easier for the everyday user</p>	<p>Flexibility is a key for They run locally and can be fast to use.</p>
Development Tools	<p>Macs are able to run swift and allow for languages like HTML and JavaScript to be used for front end and allow tools to be used such as Notebook++</p>	<p>Visual studio and eclipse can be used as IDEs for easy-to-use tools, along with notepad++, development can be used for front end development would be HTML</p>	<p>Can also run visual studio and eclipse. Offers more of a development friendly layout with usage of HTML and Javascript but can support</p>	<p>Apps can be created by using swift or android. Languages are consistent and can use HTML and JavaScript much like the other platforms.</p>

	Backend development can be used with java and python.	and Javascript much like mac. Java and python offer backend languages.	backend development still using Java, and python	
--	---	--	--	--

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:

To host Draw It or Lose it, I would use a Linux operating system on a cloud setup. Linux allows for cost friendly hosting on cloud platforms compared to windows. The cloud setup allows us to use virtual software allowing less hardware failure downtime, and more scalability for bandwidth as during peak seasons of traffic, the system can allocate more memory for demands.

Operating Systems Architectures:

Linux offers a great and safe developed kernel, compared to windows. When creating changes to the kernel, windows allow the user to run as administrator, where as Linux requires a root password for any changes at the system level. Viruses are harder to get because Linux requires the user to use a package handler to install any external packages. This protects the kernel better than what windows and mac, which safer and smarter hosting as Linux is more beginner friendly.

Storage Management:

For storage would most likely use a SSD for storage, as an HDD requires more processing time to get data into memory. The reason for this is because the disk needs to have data loaded on a spinning disk which is often times faulty and could result in a loss of data. Using a solid-state drive might be more spendy, but storing data goes smother, as it takes less time to store data, and retrieve data back to memory. For a database type I would use NoSQL as it provides an

unstructured schema, which is great for structuring storage for images as it is not concrete, so it offers an efficient way to retrieve and store data.

Memory Management:

Using Cloud a cloud base platform allows us to only pay for memory as we need it. Using instances can help the data loads would be up to the user, because they will be responsible for clearing the browsers cache. For management of how much excess data I would use ps-watcher to ensure that we will not have a crash due to a lack of memory during times when memory usage is high.

Distributed Systems and Networks:

As the server is going to be ran in the cloud outages and connectivity issues are very minimal. Because we are running the game as a website, running the game on cross platforms will not be an issue because using this method makes it easy to cross platform due to an identical online server. Using an IDE, we can export the game into the web and allow cross plays and keeping the game running at all times.

Security:

Using a role-based security system with logins and passwords can district admin functionality with player, team. And game functionality. This is important because it allows only admins to change important functionality that players will not be able to change. This protects the game because anyone who tries to change the game rules, and security features will require an admin account and password.