THE GAMING ROOM

<Name-of-Software-Application>

CS 230 Project Software Design Template

Version 1.0

Table of Contents

Document Revision History

Version	Date	Author	Comments
1.0	05/19/23	Brigitte Rollain	Information based on software design added.
1.1	06/02/23	Brigitte Rollain	Information based on operating platforms adjusted.
1.2	06/10/2023	Brigitte Rollain	Information based on memory management and storage management is updated.
1.3	06/17/2023	Brigitte Rollain	Updated recommendations information

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

Draw It or Lose will be loosely similar to the 1980s television game Win, Lose or Draw, although instead of having a player draw there will be a library of stock drawings as clues to guessing the puzzle. The game will consist of four rounds of play lasting one minute each. Drawings should be rendered at a steady rate and are fully complete at the 30-second mark. If the team does not guess the puzzle before time expires the remaining teams have an opportunity to offer one guess each to solve the puzzle within a 15-second time limit.

Requirements

A game should have the ability to have one or more teams. Each team will have multiple players assigned to it. The game and team names must be unique to allow users to check whether a name is in use when choosing a team name. Only one instance of a game can exist at any given time which can be accomplished by creating unique identifiers for each instance of a game, team, or player.

Design Constraints

- The game needs to be run on multiple platforms.
- A game can have multiple teams with multiple people.
- A check for a unique team or game name should be made.
- Only one instance of the game is allowed in memory at any given time.

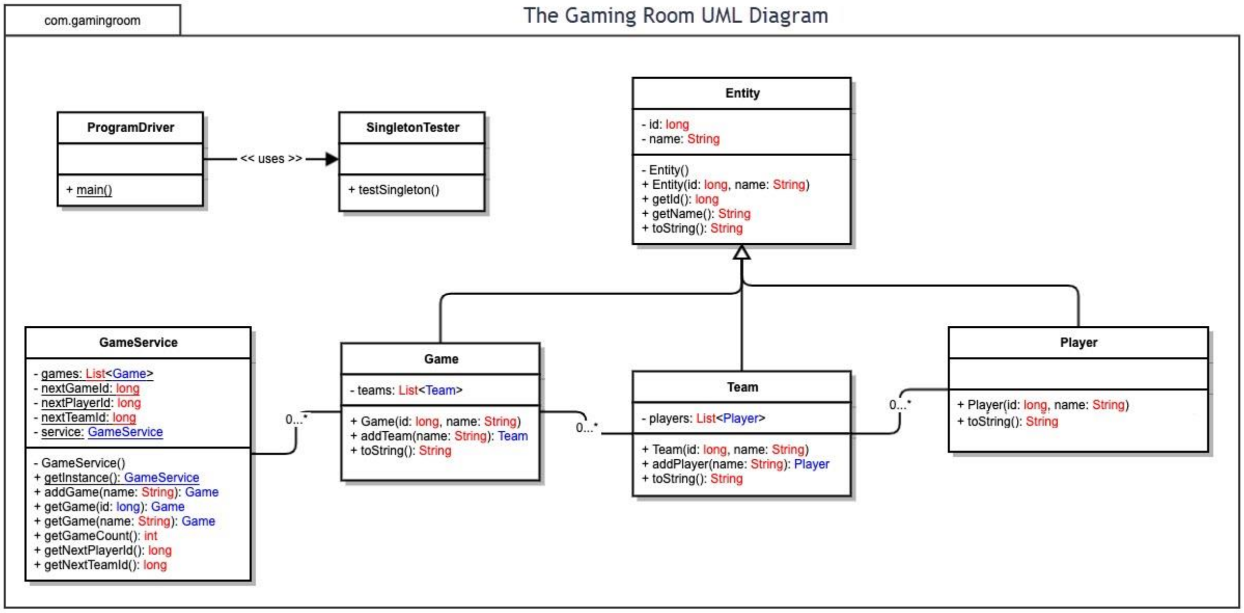
This game is currently available for android and the client would like to make it available for other platforms which may need different developers to translate the code in different languages.

System Architecture View

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

The Gaming Room UML Diagram below shows seven classes to implement in the project. The classes “GameService”, “Game”, “Team”, and “Player” all have a generalization relationship that c a zero to many relationships in descending order. That means for instance that the “GameService” class may be associated as zero or more instances of the “Game” class and each instance of the “Game” class is associated with exactly one instance of the “GameService” class. The same is true in terms of “Game” to “Team”, and “ Team” to “Player”. The classes “Game”, “Team”, and “Player” also share an inheritance relationship to the “Entity” class. This means that the code in the “Entity” class can and is used by each of the inherited classes to help reduce redundancy in the code. The class “SingletonTester” test to ensure only a single occurrence of the game is running at a time. The class “ProgramDriver” which holds the main has an associative relationship with “SingletonTester”. In this scenario, “ProgramDriver” uses what is in the “SingletonTester” class.



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 has to 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	If Mac clients are in the network, then Mac allows for extremely easy creation of features. The MacOS GUI hides complexities to make using a server somewhat easier for smaller groups. Mac servers are typically more expensive. Mac has support lines available 24/7. MacOS servers no longer exist as of April 21, 2022.	Linux may be more affordable being an open-source operating system. However, it may be that the applications and choices available to users may be overwhelming compared to Mac and Windows having recommended software. Linux servers won't randomly install new software so any maintenance is expected to be done yourself which is good to ensuring stable connection but may become tedious as software progresses. Linux, not being as popular, is less likely to be attacked by hackers. Paid versions of Linux have support lines. An expert should have little problem, but a beginner may run into many issues due to the lack of structure. Often Linux PC's hardware is customizable.	Windows often updates, which is great to stay up to date with software but may result in a lagging or crashing server. Windows is constantly being attacked by hackers as it is the most popular operating system. Windows has support lines 24/7. Windows requires minimum expertise and time as they will maintain the server's hardware and software. What is given is what is needed, they will ensure the servers cater to your needs instead of having a server that has many unnecessary tools or features given to the client. Compatibility is much easier to understand. Often Windows PC's hardware is customizable.	Mobile Devices have a large user base, are user-friendly, and less likely to be hacked. Established APP stores, like Google Play, are difficult to compete with so many companies join in and buy a server among applications in an APP store. Without relying on an APP store, mobile devices vary greatly in software and hardware which may make it difficult to generalize a project all alone. Mobile devices are also oftentimes not as powerful as laptops or PC's which limit storage and possibly also functionality. Using a server for mobile devices may make accessing the app difficult or impossible on other devices as there are functions suited solely for touch-screen operations.

Client Side	Cost should be included by using MacOS. Time should be minimal due to great connecting clients. Expertise is minimal to moderate as issues may occur but should be explained. Another issue is that Mac servers no longer exist.	Cost depends on the application. Time depends on what needs to be done in order to access the application. Expertise, moderate to high as there could be little issues if explained directly when installing the application or major faults of hardware or software without realizing.	Cost depends on the application; it may be wise to offer free trials. Time, minimal installation and minimal activation is time consuming thanks to the Windows platform being straightforward. Expertise is moderate as there may be some issues in resources and the user must know what and why there is an issue.	This can be cost effective and user-friendly as many people have phones and provides flexibility in engagement. However, revenue may have to be obtained in the infamous ad. placements rather than having consumers pay for the product, which means needing to figure out where and why to place them. Cost can be nothing to a minimal amount. Time is flexible. Expertise is also minimal and intuitive.

Development Tools	Languages included are not limited to the following: HTML, CSS, and JavaScript. Visual Studio, an IDE, should work fine with MacOS. Mac also has Visual Studio Code which is free for private and commercial use. A PyCharm license for individual use is $99 per year, for organizations it is $249 per user/per year, but the cost lowers the second year to $199 and the third year to $149.	Linux is written in C, but also supports Python, C++, Java, Perl, and more. The most well-known IDE that should work well for Linux is Eclipse, but can also use Atom, Brackets, and more. Eclipse has an Eclipse Public License (EPL) which is a commercially friendly license that allows organizations to include Eclipse software in their commercial products while asking those who create derivative works to contribute to the community. Derivative works do not include works made solely by oneself but do include works made with or modification to existing Eclipse code. A PyCharm license for individual use is $99 per year, for organizations it is $249 per user/per year, but the cost lowers the second year	Windows includes Batch, Visual basic Script, HTML, CSS, and JavaScript while also supporting C and C++ or more. Visual Studio and Eclipse as well as PyCharm are IDE's that work well with Windows. Windows also has Visual Studio Code which is free for private and commercial use. A PyCharm license for individual use is $99 per year, for organizations it is $249 per user/per year, but the cost lowers the second year to $199 and the third year to $149.	Common languages supported by mobile devices include, but are not limited to: Java, Swift, JavaScript, Objective-C, HTML5, and Kotlin. Android has IDEs that work well such as Visual Studio, Flutter, and AIDE. Cloud code IDE from Google cloud may work with multiple types of mobile devices. Eclipse has an Eclipse Public License (EPL) which is a commercially friendly license that allows organizations to include Eclipse software in their commercial products while asking those who create derivative works to contribute to the community. Derivative works do not include works made solely by oneself but do include works made with or modification to existing Eclipse code. Android Studio is free with a Freeware license. Freeware licenses do not require payment for use, but may have other restrictions, so it's

		to $199 and the third year to $149. A PyCharm license for individual use is $99 per year, for organizations it is $249 per user/per year, but the cost lowers the second year to $199 and the third year to $149.		important to understand what using this particular IDE may entail.

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**: The Windows OS would be the best operating platform offering the most flexibility and one of the lowest levels of expertise. Windows is relatively cost efficient for what they offer along with being compatible with most IDE's. Since Windows OS works with many IDEs, it should be able to code Draw It or Lose It with a popular language and translate it with minimal effort if needed. It's also the best option for customizability as a conference with representatives will be able to offer a customized package to the business.

2. **Operating Systems Architectures**: The Windows API provides services which are used by all Windows-based applications that allow access to a GUI, the system resources, and more. With Windows' layered design consisting of a user mode and kernel mode, Windows is great for non-experienced and experienced developers alike.

3. **Storage Management**: An appropriate storage management system to be used with the recommended operating platform, Windows, would be to utilize Windows OneDrive, otherwise known as Windows Cloud, to hold the software securely until it's needed to be updated or edited. The OneDrive also allows the client to pay for various sizes in storage, so Windows is willing to be flexible. The needed storage would include the gallery of images, a base code for getting and sending the images, and to hold information from the players, the gallery would be about 1.6 GB, the code just about the same, but the number of players can't be predicted, so a good starting point could be a base package of 500GB.

4. **Memory Management**: The Windows OS gives data a physical address space as well as a virtual address space. Windows also makes sure that accessing memory will not corrupt what is being accessed. For holding the library of pictures for the game, Windows may have a way to allow access to this data through the servers rather than a player having to download the entire library. Considering 200 pictures roughly sizing to 8 megabytes per image and totaling to about 1.6 gigabytes, it wouldn't be the worst thing to distribute to individuals, but if the company were to decide to update and increase the gallery or add seasonal galleries then it would really start to be a problem. This is why operators offer a variety of server packages, so all that information can be stored and retrieved from the server and the individuals don't have to worry about making room on their devices to play or access certain games. It would also make sense for the server to store information about a player, such as their ID, name, and score especially if a level system is being considered. Because memory is used to hold data that the CPU is currently processing the server can send the images intended for a round of the game all at once for the players and be held in memory, but all the images are kept in storage to be accessed when needed.

5. **Distributed Systems and Networks**: A cross-platform developmental tool supported by Windows is Unity or Unreal Engines, although Unity is more cost-friendly. Unity can support Windows, Linux, Android, and iOS making it a great choice for creating a game run in multiple platforms. A server would have to be dedicated to gaming to reduce lagging and be used to handling high traffic. Using a dedicated gaming server minimizes outages and latency.

6. **Security**: The Windows server operating system allows the user to control settings to help protect data going in and out of the server. The Windows server is often under attack being one of the most popular OS, which is why they have backups, insurance, and constantly update the servers. At the very least a backup would be provided if anything happens.