Draw It or Lose It

CS 230 Project Software Design Template

Version 1.0

Table of Contents

Document Revision History

Version	Date	Author	Comments
1.0	01/28/23	Bryan Chan	In this revision, I included the executive summary, design constraints, and the description of the project's UML 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

Our clients at The Gaming Room, currently operate a game, *Draw It or Lost It*, which as of right now is only available on Android. The game finds its inspiration from the hit 80s television game show, *Win, Lose, or Draw*. The game consists of the following sequence:

1) the application begins to render an image derived from a library of stock images
2) images are carefully drawn and are completed at the 30-sec mark
3) the game consists of four rounds, which last one minute each
4) 15 seconds are allocated for the opposing team to make their correct guess if the original team's time has expired

Our clients have requested our assistance in helping to develop their application. Their request is for a web-based version of the *Draw It or Lost It game*, which can accommodate multiple platforms. Besides this, they have asked that we implement the option for the game to include more than one team, for each team to have multiple players, each game and team name must be unique, and that only one instance of each game can be created.

To accomplish their request, we will create a web-based version of their game, powered by a cloud-computing platform in order for the program to be run on multiple devices. The program will utilize a third-party library containing stock images, in order to render the images.

Requirements

< Please note: While this section is not being assessed, it will support your outline of the design constraints below. *In your summary, identify each of the client's business and technical requirements in a clear and concise manner.>*

Design Constraints

After looking through the client's requests and requirements, we have discovered three design constraints that will be present throughout the software's lifecycle. First, the application must be web-based. This will affect our decision on which language and server to choose. Second, the app must utilize images from a third-party stock library. We will have to ensure that the program successfully connects with the library, so that the program can utilize the stock images. Third, there can only be one instance of a game. To accommodate this, the app could benefit from the singleton design pattern, which accomplishes this purpose.

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.

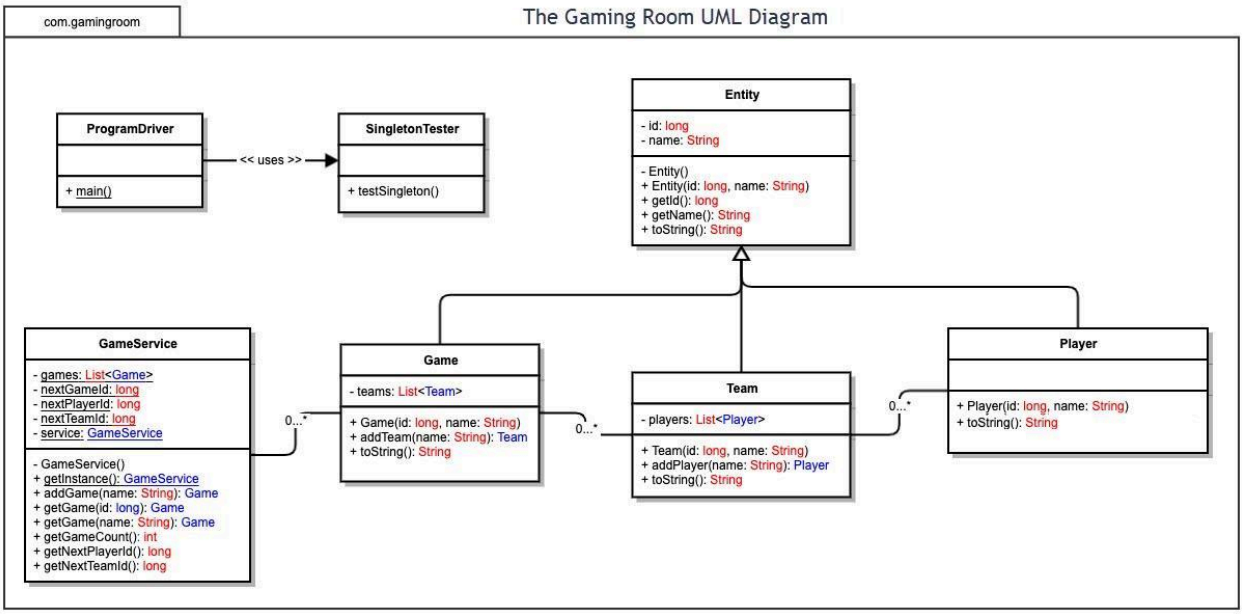
Domain Model

Fundamental principles of the object-oriented programming paradigm can be seen in the following UML class diagram. For starters, the SingletonTester class is a composition of the ProgramDriver class.

Next, the Entity superclass, has id and name as attributes, which will be inherited by the Game, Team, and Player subclasses. Within the Entity superclass, there is an empty private constructor, so that the Entity class can accommodate the singleton design pattern, so that only one instance of Entity gets created. This is followed by the getId() and getName() getter methods. Lastly, there is an toString() method to print out the contents of the Entity class.

The GameService class holds the following attributes: games (list), nextGameId, nextPlayerId, nextTeamId, service (GameService). Along with those attributes, the GameService class also includes the following methods: getInstance(), addGame(), getGame(), getGameCount(), getNextPlayerId(), getNextTeamId(). Within the addGame() method, the iterator pattern is used.

In all the Game, Team, and Player subclasses, polymorphism is demonstrated since they are inheriting the attributes and methods from the Entity class, and overrides them so that it can print out their values. Since they inherit from the superclass, they also demonstrate inheritance. Within the Team and Game classes, the iterator pattern is used (addTeam(), addGame()). The iterator pattern is important, as it will ensure that only unique teams/games are added.



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	The benefits of using macOS as the operating system for hosting web applications, is that it has an "easy administration, great support, and comes with an easy workload distribution" (Marijan, 2022). However, it can only run on Apple products which can be pricey, not to mention the lack of third-party applications (Marijan, 2022).	The benefits of using Linux as the operating system for hosting web applications, is that it includes "high security, UNIX, and a wide variety of distributions" (Marijan, 2022). However, it does have a lack of long term support for some distributions, and can require extensive technical knowledge (Marijan, 2022).	The benefits of using Windows as the operating system for hosting web applications, is that it has "[...]support for symmetric multi-processor systems, great third-party application support, and many versions to choose from[...]"(Marijan, 2022). However, compared to other operating systems, it has more security vulnerabilities (Marijan, 2022).	The benefits of using a mobile operating system for hosting web applications, is that it has "the ability to do multiple tasks, stability, and secure mobile data access" (Bhatnagar, 2023). However, "development is harder for developers and can run slowly for low spec smartphones" (Bhatnagar, 2023).

Client Side	Developing on MacOS will require great investment in equipment as MacOS applications have to be built with pricey Apple products, lots of time to look for qualified macOS developers, and great expertise in the macOS platform (Marijan, 2022).	Developing on Linux requires little investment in equipment since Linux is primarily open sourced (Gewirtz, 2019). In addition, Linux apps will take a long time to develop, as the Linux OS typically takes a while to configure (Marijan, 2022).. Because of the complexities of the Linux distributions, it will require a knowledgeable Linux developer who knows the ins and outs of the operating system (Marijan, 2022).	Developing on the Windows platform will require little investment in equipment since Windows is widely available on a number of devices (Gewirtz, 2019). In addition, many tools and applications are available for Windows, which reduces the overall time of development (Gewirtz, 2019). Finding a Windows developer is also not difficult, as the operating system is one of the most popular (Gewirtz, 2019).	Developing for a mobile operating system requires greater investment in specialized mobile developers, expertise in the mobile OS platform, and a greater time of development Bhatnagar, 2023).
Development Tools	Just like iOS apps, macOS applications needs to be built on its respective IDE—XCode. In addition, it will have to be built with the Swift language.	Developing applications for the Linux operating system can either be built with "C, C++, Python, *etc.*" (Bolton, 2015). Because of Linux's flexibility with operating systems, it can be built with a variety of IDEs such as Visual Studio and Jetbrains IntelliJ.	Developing applications for Windows is typically done with C++ and C# (QuinnRadich et al., 2023). This is can be done on the Visual Studio IDE (QuinnRadich et al., 2023).	To deploy a mobile application, you will needed to utilize an IDE that is specific to your platform of choice (Android Studio for Android OS or XCode for iOS). The languages will be different as Android apps are mostly built with Java or Kotlin, while iOS apps are built with Swift.

References

Bhatnagar, S. (2023, October 4). *Different types of operating System [Pros and cons]*. KnowledgeHut: Professional Bootcamps and Certification Courses Provider for your Future. https://www.knowledgehut.com/blog/web-development/types-of-operating-system

Bolton, D. (2015, March 9). *Best programming languages for Linux Devs*. Dice Insights. https://www.dice.com/career-advice/best-programming-languages-linux-devs

Gewirtz, D. (2019, April 8). *Windows, Mac, or Linux? We compare the pros and cons of these computing platforms*. ZDNET. https://www.zdnet.com/article/windows-mac-or-linux-we-compare-the-pros-and-cons-of-these-computing-platforms/

Marijan, B. (2022, March 10). *Server operating system: Server OS types & how to choose*. Knowledge Base by phoenixNAP. https://phoenixnap.com/kb/server-operating-system

QuinnRadich et al. (2023, October 13). *An overview of app development*

option - Windows apps. Microsoft Learn: Build skills that open doors in your career. https://learn.microsoft.com/en-us/windows/apps/get-started/?tabs=winappsdk-winui%2Cnet-maui

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**

   For this project, I recommend going with the Windows operating platform. The reason is that Windows will allow the application Draw It or Lose It to be accessible to other computing environments.

2. **Operating Systems Architectures**

   According to the article, “Server Operating Systems: Server OS Types & How to Choose” from Bosko Marijan of phoenixnap.com (2022), one of the benefits of using the Windows platform, is that it has “[...]support for symmetric multi-processor systems, great third-party application support, and many versions to choose from[...]"(Marijan, 2022). In addition to this, the platform requires little investment since the Windows platform is widely available.

3. **Storage Management**

   For our client’s storage needs, I would recommend going with a cloud storage option. The reason why resides on various factors: 1) size and 2) cost. For starters, going with a cloud storage option allows flexibility with the storage size. Because of this, the price is often based on how much storage is used.

4. **Memory Management**

   Using a memory allocation tool within the program, will maximize the program’s overall memory efficiency. In specific terms, the program should utilize an automatic allocation process. According to Robert Sheldon in the techtarget.com article, “Memory Management” (2022), the automatic allocation process can be done within the programming language itself (Sheldon, 2022). Using an automatic allocation process will help the program distribute the memory wherever it is most needed.

5. **Distributed Systems and Networks**

In order for the application to communicate with various platforms, a distributed system will need to be implemented. For the purposes of this application, we will be utilizing the client-server system. According to Chrissy Kidd in the Splunk article, " Distributed Systems Explained" (2023), "Client-server systems, the most traditional and simple type of distributed system, involve a multitude of networked computers that interact with a central server for data storage, processing or other common goal." (Kidd, 2023). This model will allow frequency communication between various platforms, which will enable the app to be used in various platforms.

6. **Security**

According to Bosko Marijan in the phoenixnap.com article, "Server Operating Systems: Server OS Types & How to Choose" (2022), compared to other operating systems, Windows has more security vulnerabilities (Marijan, 2022). Because of this, security features such as data encryption and two-factor authentication need to be added in order to protect the data exchange of the user's information.

References

Kidd, C. (2023, August 29). *Distributed systems explained*. Splunk-Blogs. https://www.splunk.com/en_us/blog/learn/distributed-systems.html

Marijan, B. (2022, March 10). *Server operating system: Server OS types & how to choose*. Knowledge Base by phoenixNAP. https://phoenixnap.com/kb/server-operating-system

Sheldon, R. (2022, June 6). *What is memory management in a computer*

environment? techtarget.com.

https://www.techtarget.com/whatis/definition/memory-management