

**The Gaming Room**

# Software Design Report

Version 1.2

## Table of Contents

[**Software Design Report**](#_Toc115077317)

[**Table of Contents 2**](#_Toc115077318)

[**Document Revision History 2**](#_Toc115077319)

[**Executive Summary 3**](#_Toc115077320)

[**Requirements 3**](#_Toc115077321)

[**Design Constraints 3**](#_Toc115077321)

**System Architecture View 4**

[**Domain Model 4**](#_Toc115077324)

[**Evaluation 5**](#_Toc115077325)

[**Recommendations 7**](#_Toc115077326)

## 

## [Document Revision History](#_grjogdjh5fi8)

| Version | Date | Author | Comments |
| --- | --- | --- | --- |
| 1.0 | 05/21/2023 | Rayyan Abdulmunib | The document’s cover page, Revision History, Executive Summary, Design Constraints, System Architecture View, Domain Model, and Recommendations sections all were updated. |
| 1.1 | 06/04/2023 | Rayyan Abdulmunib | The document’s Revision History, Executive Summary, and Evaluation sections all were updated. Design constraints were added. |
| 1.2 | 06/18/2023 | Rayyan Abdulmunib | The document’s Revision History, Evaluation and Recommendations sections were updated. |

## [Executive Summary](#_sbfa50wo7nsh)

The information in this report will be used to make recommendations for the Gaming Room project. Based on the present game, “Draw It or Lose It”, which is exclusively available on Android, the Gaming Room project aims to create a web-based game that serves numerous platforms. The goal of the game is for various teams of several players to compete in four rounds of one minute each. When a photo is chosen from a library, one team guesses till time runs out. If no one responds, each opposing team member has 15 seconds to respond before time runs out.

## Requirements

* More than one team can participate in the game at a time.
* Each team should consist of multiple players.
* Every team and game name must be unique to check if the name already exists.
* Only one instance or session of the game can exist in memory at a time.
* The web version of the game should closely mimic the Android version of the game, Draw It or Lose It. It is like the 1980s television game “Win, Lose, or Draw”.

## Design Constraints

* A website application must be built.
* A hardware server machine or cloud-based equivalent must be provided along with an accompanying operating system that will host the website.
* Sufficient storage is needed to hold a large library of drawings.
* Authorization/Authentication: Mechanism for login/security.

## [System Architecture View](#_ilbxbyevv6b6)

## [Domain Model](#_8h2ehzxfam4o)

The Entity class establishes a relationship between the Game, Team, and Player classes. This implies that they all inherit or get information from the Entity class. We can demonstrate this via inheritance in the UML diagram. The Entity class becomes a superclass. So, when we look at their relationship, we can observe that the Team and Player classes are of the "has a" type. A Game has a Team, while a GameService has Games. Aggregation (“has-a”) is what we name it in UML diagram. Furthermore, what is meant by a user "has a" is that it is an instance of one class and a reference to an instance of another class. Additionally, the 0-to-many relationship between two classes is denoted by the “0..\*” in the UML diagram below. For instance, A Game (Game class) can have 0+ or 0 to many Teams (Team class instances) just like a Team (Team class) can have 0 to many Players (Player class instances), etc. Nonetheless, looking at the UML diagram below, we can see that the GameService class has a reference to the Game class, the Game class has a reference to the Team class, and the Team class has a reference to the Player class.

**"The Gaming Room UML diagram. The top of the diagram is labeled as com dot gamingroom. Test boxes are placed in two layers. The first layer has three text boxes and the second layer has four of them. In the first layer, the 'ProgramDriver' textbox points to 'SingletonTester' textbox. The 'ProgramDriver' textbox contains the text 'asterisk main round brackets.' The 'SingletonTester' textbox contains the text 'asterisk testSingleton round brackets.' The arrow between these two text boxes are labeled 'open two angle brackets uses close two angle brackets'. In the second layer, there are 'GameService', 'Game', 'Team', and 'Player' text boxes. The 'GameService' textbox has texts arranged in two layers. The first layer contains games colon List open angle bracket Game close angle bracket, nextGamesId colon long, nextPlayer Id colon long, nextTeamId colon long, and service colon GameService. The second layer contains GameService round brackets, getinstance round brackets colon GameService, addGame open parenthesis name colon String close parenthesis colon Game, getGame open parenthesis id colon long close open parenthesis colon Game, getGame open open parenthesis name colon String close open parenthesis colon Game, getGameCount round brackets colon int, getNextPlayerID round brackets colon long, and getNextTeamId round brackets colon long. The 'GameService' box is connected with the 'Game' textbox with a line labeled 'zero dot dt dot asterisk'.  The 'Game' textbox also contains text in two layers. The first layers contains the text teams colon List open angle bracket Team close angle bracket. The second layer has Game open round bracket id colon long comma name colon String close parenthesis, addTeam open parenthesis name colon String close parenthesis Team, toString round brackets colon String. The 'Game' textbox is connected with the 'Team' textbox with a line labeled 'zero dot dt dot asterisk'. The 'Team' textbox also contains text in two layers. The first layers contains the text players colon List open angle bracket Player close angle bracket. The second layer has Team open parenthesis id colon long comma name colon String close parenthesis, addPlayer open parenthesis name colon String close parenthesis colon Player, and toString round brackets colon String. The 'Team' textbox is connected with the 'Player' textbox with a line labeled 'zero dot dt dot asterisk'. It contains the text Player open parenthesis id colon long comma name colon String close parenthesis and toString round brackets colon String. The 'Game', the 'Team, and the 'Player' boxes point to the 'Entity' textbox in first layer. The 'Entity' textbox contains text in two layers. The first layer has the text id colon long and name colon String. The second layer has Entity round brackets, Entity open parenthesis id colon long comma name colon String close parenthesis, getId round brackets colon long, getName round brackets colon String, toString round brackets colon String.**

## [Evaluation](#_2o15spng8stw)

| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| --- | --- | --- | --- | --- |
| **Server Side** | Mac has simple access and server configuration. Graphical User Interface (GUI) that is simple to use. Terminal commands that are adaptable. Proprietary software (Apple) and higher-end hardware required. High costs for licensing. Stable housing sites. | Cost effective. The platform is difficult to navigate. Command shell for easy server configuration and access. Open-source software and legacy hardware. Low costs for licensing. Stable housing sites. | The server side is costly. A user-friendly GUI. There is a command prompt. Proprietary software (Microsoft) and higher-end hardware required. High costs for licensing. Unstable housing sites. | Other devices have better specifications. The characteristics of mobile devices differ from one user to the next. |
| **Client Side** | Users pay a high price. A moderate amount of time and experience is required. Accurate skills are required to navigate the operating system (OS). Low software compatibility. Easy and free software installation. Good security. Proprietary software and product is required with Mac OS. Pre-installed web browser Safari is the most used browser. Ability to install and use other browsers like Google Chrome, Microsoft Edge, Firefox, etc. | There is a significant amount of knowledge and time necessary. To use the operating system, Linux data is necessary. The most expensive option for Linux users. Limited software compatibility. Difficult software installation. Most secure. Mozilla Firefox is the default browser and most used browser. Ability to install and use other browsers like Google Chrome, Microsoft Edge, Firefox, etc. Difficult with Mac OS browser. | Costs more than Linux systems. Setup is simple to learn and understand. Expertise is not required. High software compatibility. Slightly difficult software installation and costly. Low security. Internet Explorer comes as default, pre-installed, and most used web browser. Ability to install and use other browsers like Google Chrome, Microsoft Edge, Firefox, etc. Works well with every web browser due to extensive use of open-source software. | Provide flexibility to  clients or even  developers to see  updates at any  place. Slightly more  difficult to  implement than  other devices.  Allow clients or even developers to view updates from any location. Implementation is slightly more complicated than with other devices. More responsive design/UI. Fast load times. Offline functionality. More secure. Better scalability. Easier to maintain. Safari browser for Iphones/iOS and Google chrome browser for most Androids. Ability to install and use other browsers like Google Chrome, Microsoft Edge, Firefox, etc. Difficult with most browsers. |
| **Development Tools** | HTML, CSS, Xcode and JavaScript are examples of programming languages used. Frontend development libraries are available. PyCharm, GitHub, Visual Studios, and other development tools are also available. Deployment may be costly with licensing costs associated both with the development tool and Mac app store. Requires developer account. | HTML, CSS, and JavaScript are examples of programming languages used. Frontend and language libraries are provided. Development tools like JavaScript, Ruby, PHP, and Python are all available on Linux platforms. Deployment may be costly with licensing costs associated with the development tool. | HTML, CSS, and JavaScript are examples of programming languages used. Frontend and language libraries are provided. Eclipse, command prompt, PyCharm, Eclipse, and other developer tools are available. Deployment may be costly with licensing costs associated both with the development tool and Windows store. Requires developer account. | HTML, CSS, Swift, Java and JavaScript are examples of programming languages used. Frontend and language libraries are provided. PHP, C++, and Python are examples of development tools that are available for use. Development and deployment may depend on the mobile OS (Android or IOS). Requires developer account. The development team may need to be equipped with both Android and IOS expertise. |

## 

## Recommendations

1. **Operating Platform**: Microsoft Windows is the best operating system for The Gaming Room to spread "Draw It or Lose It" to different platforms and devices. Windows is the best option because of the following:
   * There are plenty of IDEs to choose from.
   * It is user-friendly.
   * Easy to set up and learn.
   * As of January 2023, Windows is the most commonly used operating system at around 70.39% people using it around the world (Taylor, 2023).
   * Has a wide range/variety of open-source development tools to develop the game in.
2. **Operating Systems Architectures**: Microsoft Windows is a graphical operating system that was created and distributed by Microsoft. It allows you to store files, run software, play games, watch videos, and access the internet. The architecture comprises HAL, driver, microkernel, executive Services. In Kernel mode, the executing code has complete and unrestricted access to the underlying hardware. It can execute any CPU instruction and reference at any memory address. Kernel mode is generally the most trusted function of the operating system (Windows Architecture explained by Stacksol, 2018).
3. **Storage Management**: Storage sense is a useful feature in Microsoft Windows. This allows you to examine and manage files on your hard disk, as well as how much space they use. Generally, Windows comes with more storage space than many major competitors like Mac OS. Windows operating systems have storage management utilities such as Disk Cleanup and Disk Defragmenter. Moreover, third-party apps such as CCleaner and Glary Utilities can also be used to manage storage with ease. Because Windows machines are built to function on a larger range of hardware, they are more adaptable and generally come with greater storage space.
4. **Memory Management**: The storage sense in Windows would allow for the storing and control of “Draw It or Lose It” photographs and game players. It also helps you to keep them all together in a safe place in your memory. Draw It or Lose It is a graphics-heavy program that demands a lot of memory to run smoothly. Windows employs a number of memory management strategies to guarantee that the software works quickly and does not consume excessive memory, which might impair the performance of other applications.

* When physical memory (RAM) is restricted, Windows uses virtual memory to increase the amount of available memory. When running memory-intensive apps like Draw It or Lose It, virtual memory allows the operating system to use hard disk space as an extension of RAM, which can help to increase performance.
* Memory compression is a function in Windows 10 that compresses data stored in memory to free up space. This can aid in reducing the amount of memory utilized by Draw It or Lose It and other applications.
* Memory prioritizing is utilized by Windows to assign memory resources to active apps. This means that if Draw It or Lose It is the current application, it will be prioritized in terms of memory allocation, which might assist in improving speed.
* Windows has a built-in software called Task Manager that may be used to monitor application memory utilization and terminate processes that are using too much memory. This can help free up memory resources for applications like Draw It or Lose It.

1. **Distributed Systems and Networks**: A distributed software architecture can be employed to allow Draw It or Lose It to connect across platforms. This design entails breaking down the software into smaller components that may be deployed and operate on various platforms while connecting with one another to provide the desired functionality. To achieve this, the various components of Draw It or Lose It can be designed as microservices, which are compact, modular, and self-contained services. Each microservice can run on a separate platform or server and connect with one another via APIs or message queues. A network infrastructure is necessary to permit communication between the various platforms, taking into account many considerations, this network infrastructure can be constructed to support the requirements of Draw It or Lose It considering factors like connectivity, outages, and security.
2. **Security**: Microsoft Windows includes security software by default. It has a number of security features and capabilities that can be utilized to protect user data on and between platforms. Draw It or Lose It may protect and secure user information by utilizing encryption, authentication and access control, network security measures, secure protocols, and regular updates and patches. Each of these is explained below.

* Encryption: Encrypting user information is one of the most effective ways to protect it. Encryption is the process of transforming user data into a code that can only be deciphered with a decryption key. Encryption solutions in Windows include BitLocker for full-disk encryption and Encrypting File System (EFS) for file-level encryption.
* Authentication and Access Control: Mechanisms for authentication and access control can be used to limit access to user information. Passwords, biometric authentication, and smart cards are among the authentication methods available in Windows. Access control mechanisms can be used to limit user data access depending on characteristics such as user roles and permissions.
* Network Security: Mechanisms such as firewalls, intrusion detection and prevention systems, and virtual private networks (VPNs) can be used to safeguard user information in transit across different platforms. Windows Firewall is a built-in firewall in Windows that may be customized to block network access.
* Secure Protocols: It is critical to utilize secure protocols such as HTTPS and SSL/TLS when connecting between platforms. These protocols encrypt data in transit and authenticate the communication parties. These protocols are supported by Windows and can be configured to utilize them.
* Regular Updates and Patches: It is critical to apply updates and patches on a regular basis to guarantee that Windows is secure and protected against known vulnerabilities. These upgrades can fix known security concerns and improve the platform's overall security.

**References**

Taylor, P. (2023, February 27). *Computer Operating Systems Market Share 2012-2023*. Statista. Retrieved from <https://www.statista.com/statistics/268237/global-market-share-held-by-operating-systems-since-2009/#:~:text=Microsoft’s%20Windows%20is%20the%20most,OS%20market%20in%20January%202023>

*Windows Architecture explained by Stacksol*. (2018, January 08). <https://www.slideshare.net/Stacksol/windows-architecture-explained-by-stacksol>