7-1 Project Three Submission

Craig A Nelson

Department of Computer Science, Southern New Hampshire University

CS-230-T4461 Operating Platforms 22EW4

Professor Sean Morris

April 17th, 2022

Icon

Description automatically generated

**Draw It or Lose It**

**CS 230 Project Software Design Template**

**Version 1.0**

Table of Contents

SchoolHeader…………………………………………………………………………………………………………………………………....1

CS 230 Project Software Design Template  [2](#_Toc101363073)

[Table of Contents 3](#_Toc101363073)

[Document Revision History 3](#_Toc101363073)

[[Executive Summary 4](#_Toc101363073)](#_Toc101363073)

[UML Diagram 5](#_Toc101363073)

[Evaluation 6](#_Toc101363073)

Evaluation………………………………………………………………………………………………………………………………………….7

Evaluation 8

Recommendations 9

Recommendations ………………………………………………………..……………………10

References…………………………………………………………………………………...…11

[**Document Revision History**](#_grjogdjh5fi8)

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| 1.0 | <03/20/22> | Craig A Nelson | This is the original version |

## [Executive Summary](#_sbfa50wo7nsh)

The structure of the game is having multiple teams consisting of 2 or more players. The players on a team get to participate for four rounds at a minute each. Drawings are rendered at a steady rate and are fully complete at the 30-second mark. If that team is unable to guess the puzzle before time expires. Remaining teams will have 15 seconds to solve the puzzle.

The Gaming Room wants to develop a web-based game called “Draw It or Lose It”. It is available on the Android platform presently. The Game Room is requesting the game to play on multiple platforms mainly Windows, Linux, and Apple (Mac and iOS) devices. The app will need:

* One or more teams involved
* Each team has multiple players
* Game and Team names must be unique to allow users to check whether the name is in use or free
* Only one instance of the game can exist at any time.

We can write the code in a universal language (Java- WORA(write once run anywhere).

**Design Constraints**

**Domain Model**

Entity is base class or a superclass and has a relationship between Game, Team, and Player class. This means Game, Team, and Player classes all inherit and get information from Entity. The UML shows the inheritance. The “name” and “id” are common throughout the classes. Looking at (HAS-A) aggregations in the UML. We see Team and Player is a “has a” type. While Game “has a” Team and GameService “has a” Games. When a user “has a” I mean it's an instance of one class and has a relationship to an instance of another class. When we look at this diagram, we see GameService has a reference of Games, Games a reference of Team, and Team a reference of Player.

**"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.**

| **Development Requirements** | **Mac** | **Linux** | **Windows** | **Mobile Devices** |
| --- | --- | --- | --- | --- |
| **Server Side** | The macOS Server **pros** are easy administration, intuitive GUI, great support, and easy workload distribution across multiple machines. Easy to increase the processing power. The OS comes with an unlimited user license.  The **cons** are that macOS Server only runs on Apple hardware, expensive $$$, and not as many third-party applications. Also, while Apple implements open-source software in its system, there are changes specific to macOS, requiring working around some issues not present in Linux. (phoenixNAP,2022) | Linux is a family of UNIX-like operating systems It is open-source. Has many distributions: Ubuntu Server. Debian Server.Fedora.  OpenSUSE Leap.  SUSE Linux Enterprise Server. Arch Linux.  The **pros** of a Linux server OS are high security, integrated open-source software, including high-level language compilers, and the ability to control the system using a GUI.  The **cons** are the lack of long-term support for some distributions and certain complex operations, such as system updates. (phoenixNAP,2022) | The **pros** of a Windows server OS are the easy to use GUI, support for multi-processor systems, great third-party application support, will give 10 year support and many versions to choose from.  The **cons** are the need for user-based licensing and more virus security threats compared to other platforms.  (phoenixNAP,2022) | Needed are to connect the mobile device to a database on a compatible server. That will have to be coded by hand. Time-consuming and difficult. If you need a great deal of customization the team can consider it.  (phoenixNAP,2022) |
| **Server Side Listening Costs** | The servers are expensive. **Mac** servers are not commonly used for game servers. | **Linux** Server OS’s has free licenses available that would work just fine. There are minimal-cost licenses on some distributions. | Server Side for **Windows** does have licensing costs associated with it. They have their own OS’s Windows: Here are the last Three versions:  Windows Server 2022  (latest)  Windows Server 2019  Windows Server 2016 | **Mobiles** are not servers. They would have to be connected to a database. |
| **Actual Cost** | Add macOS Server to your Mac from the Mac App Store for just  **$19.99.**  (Apple, n.d.). | For web servers, the two most popular versions are Ubuntu and CentOS. Both are open-source and **FREE**  (phoenixNAP2022) | For Standard Windows Server 2022  **$1069**  (Microsoft, 2022) | **N/A** |
| **Client Side** | Response to Client-side HTML, JavaScript and CSS-  Responsive.  Displayed in Browser.  One choice is Safari is Mac’s Browser included(free) with OS | Response to Client-side HTML, JavaScript and CSS-  Responsive.  Displayed in Browser.  Firefox is Linux’s default Browser included(free) with OS | Response to Client-side HTML, JavaScript and CSS-  Responsive.  Displayed in Browser.  Windows uses Edge or Internet Explorer by default. (free) | Response to Client-side HTML, JavaScript and CSS-  Responsive.  Browsers vary but all are free for iOS, android or windows. |
| **Development Tools** | There will be a lot of overlap when discussing development tools for these platforms. As mentioned before Java is a WORA (Write once run anywhere). IDEs for Java are Visual Studio Code, IntelliJ and Eclipse are available for MAC as well as other OS’s. but most languages can be developed on a Mac. | Linux has great support for most programming languages. You can write in several languages. IDEs for Java are Visual Studio Code, IntelliJ and Eclipse are available for Linux as well as other OS’s | Windows is possibly the most versatile of any of OS’s. The top IDEs for Java are Visual Studio Code, IntelliJ and Eclipse are available for MAC as well as other OS’s. There are many tools and third-party tools in Windows to get whatever programming job done. | Java is the official language for Android App Development and so it is the most used language as well. It will work for Windows phones and iOS. |
| **IDE COST**  **FREE** | Sublime Build 4126  Visual Studio Code  Eclipse IDE 2022‑03 | Sublime Build 4126  Visual Studio Code  Eclipse IDE 2022‑03 | Sublime Build 4126  Visual Studio Code  Eclipse IDE 2022‑03 | N/A |
| **Expertise** | Server-side written in Java,  Response to Client-side HTML, JavaScript and CSS- Responsive | Server-side written in Java,  Response to Client-side HTML, JavaScript and CSS=  Responsive | Server-side written in Java,  Response to Client-side HTML, JavaScript and CSS-  Responsive | Server-side written in Java,  Response to Client-side HTML, JavaScript and CSS-  Responsive |
| **Time** | varies | varies | varies | varies |

**Recommendations**

**Operating Platform:**

After very careful consideration. We recommend a Cloud Platform. The Cloud platform that we chose is Amazon Web Services. Because you have 1000 teams, four players on each team. Your use of resources will be substantial. Choosing AWS will alleviate many issues in the development process by automating some of the tasks.

“It is one of the best choices for multiplayer games.  
…secure cloud platform.  
…game servers are safe with DDoS protection and security from frequent network attacks.  
…deliver optimum performance, lowest latency, and greater cost savings.”( Amazon Gamelift Service, n.d.)

**Operating Systems Architectures**:

Above we narrowed are choices to Linux and Windows. Mac was just to expensive. We recommend Linux server the cost savings are substantial.

|  |  |  |
| --- | --- | --- |
| **Linux** | | |
| Instance Type | Hourly Spot | \*Monthly Spot w/ Autoscaling (365 hours) |
| m5.large | $0.02412 | $8.80 per Month |

|  |  |  |
| --- | --- | --- |
| **Windows** | | |
| Instance Type | Hourly Spot | \*Monthly Spot w/ Autoscaling (365 hours) |
| m5.large | $0.1344 | $49.06 per Month |

**Storage Management:**

The Game required storage for 200 images at 8MBs per image. Rounded up that is 2GBs. You will want to store the program which is minimal. User information and Game statistics which could add some storage. With the lowest program from AWS will cover 5GB. Although that would be enough. We would recommend the next step at 10GB to start. As for traffic it is based “pay as you go”. We ran a sample scenario and received a sample quote -see attached pdf.

**Memory Management:**

AWS has many options available. Here is a use case sample:

“R5 instances are well suited for memory intensive applications such as high performance databases, distributed web scale in-memory caches, mid-size in-memory databases, real time big data analytics, and other enterprise applications.”(Daly,1987)

We will have to work together to define what option you would like to start with.

**Distributed Systems and Networks:**

What makes AWS GameLift attractive for us. Is that it manages the servers across 22 regions on 5 continents. That is a network that we will tap into when we associate our clients with AWS. Again we ”pay as you go”, we can avail of the matchmaking service, it follows the rules we define. AWS uses TCP and UDP with network stacking. We can also take advantage of low latency.

**Security:**

AWS GameLift Security uses a shared responsibility security system. When you configure the account you will pick out a security service and configure it to your needs. This is on top of what AWS GameLift will secure On their platform. The schematic below should make clear The Gaming Room’s responsibility.

Diagram

Description automatically generated

\*Note it seems like we are pushing responsibilities to the Cloud. This is modern day game development. We should embrace it now rather than later.

**References**

9 best game server hosting platforms for everyone. Geekflare. (2021, June 21). Retrieved April 23, 2022, from <https://geekflare.com/game-server-hosting/>

Amazon Gamelift Service - docs.aws.amazon.com. AWS Amazon Gamelift Service - docs.aws.amazon.com. (n.d.). Retrieved April 24, 2022, from https://docs.aws.amazon.com/gamelift/latest/apireference/API\_GameServerGroupAutoScalingPolicy.html

Centos vs ubuntu: Choose the best OS for your web server. phoenixNAP Blog. (2022, March 17). Retrieved April 8, 2022, from   
<https://phoenixnap.com/blog/centos-vs-ubuntu#:~:text=For%20web%20servers%2C%20the%20two,free%20community%2Dsupported%20operating%20systems>.

Daly, D. J., &amp; Daly, D. J. (1987). Economics 2: EC2. Amazon. Retrieved April 24, 2022, from https://aws.amazon.com/ec2/instance-types/

MacOS server. Apple. (n.d.). Retrieved April 8, 2022, from <https://www.apple.com/macos/server/>

Server operating system: Server Os Types &amp; How to choose. Knowledge Base by phoenixNAP. (2022, March 10). Retrieved March 22, 2022, from   
<https://phoenixnap.com/kb/server-operating-system>

Windows server 2022 licensing &amp; pricing. Microsoft. (n.d.). Retrieved April 8, 2022, from https://www.microsoft.com/en-us/windows-server/pricing