<Draw it or Lose It!>

# Software Design Template

Version 1.0

## Table of Contents

[Software Design Template 1](#_Toc21086455)

[Table of Contents 2](#_Toc21086456)

[Document Revision History 2](#_Toc21086457)

[Executive Summary 3](#_Toc21086458)

[Requirements 3](#_Toc21086459)

[Design Constraints 3](#_Toc21086459)

[Rationale 3](#_Toc21086460)

## Document Revision History

| Version | Date | Author | Comments |
| --- | --- | --- | --- |
| 1.0 | 07/15/2023 | Chase Carter | Executive Summary |
| 2.0 | 07/26/2023 | Chase Carter | Requirements |

## 

## Executive Summary

“Draw It or Lose It!” is a game for the client “The Gaming Room”. The client is asking that the game be moved from an android-only game to being compatible with Windows, Mac, Linus, and IOS operating systems. The goal is to make the game cross platform between all these operating systems.

## Requirements

The developers must be able to use react to make the game allow cross-platforming. The developers must also make the APIs compatible with React DOM. API is the application programming interface that allows the software of the game to connect between different servers to be able to get the requests of the user. React Dom is an element of React that connects the user interface with the application. This can be used for all the different applications that will be in play for the game as well as the web-based parts.

## Design Constraints

The current design constraint is that the game is compatible with only Android applications. This is currently in JavaScript, CSS, and HTML. The developers will use this to be able to be compatible with React, this will take some time. The time that it takes for the developers to work on this could be lengthy depending on how many developers are working on the game. There may also be some constraints by having issues with the cross-platforming capabilities. Cyber security is another design constraint that is possible. The application may need to store the user’s information. Potential threats to security will have to be addressed in the application with a cyber security team or with the developers. A suggestion is that the users have a dual log-in method that protects them better. This is by having a basic username and password and then using a phone number authentication. The game must be compatible with all web browsers, meaning it must have its own domain and servers linked to the web format.

## Rationale

The application will be “Cross-Platform” as stated earlier, this means that it will function in all OS and will be a web-based server. The objective is that the front end will be an application that can be accessed on mobile devices. This will be the most effective deployment due to the accessibility of the users. This will be done with React and will use web-based servers for the application programming interface. This is how the application will operate. For the server, the recommended will be Mac and the IDE that is recommended is Xcode. This is recommended due to the large capacity of memory in the IOS operating system. This can be accomplished with any of the OS and IDE but issues may arise due to IOS applications can transfer easily to the other OS but it is much harder to do the opposite. For example, if the developers make an application in IOS it will be much easier to transfer to Android for mobile devices. If it is made in Android there will be many hurdles that the developers will have to jump through to make it compatible with IOS. The user's information can be stored in a cloud-based technology such as Amazon Web Services, but for security, the recommendation is that the users have a dual authentication method to further protect their information. Cyber security support is recommended, this will aid in the protection of valuable information and server stability.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Requirements | IOS | Linux | Windows | Mobile Devices |
| Server Side | To keep costs down to run the server, “WebSocket in HTML”. This allows the client to connect to the browser. The API mentioned earlier will need to connect to something, I would recommend using “flask”.  IOS has bult in server tools inside the OS that can be utilized. These can be written in “Python”. However only specific hardware can run it, this is a con because it will be limited to the hardware. | To keep costs down just like IOS, the API I would recommend is “flask”. Something that works great with Linux is using different cloud technologies. “Amazon Web Services” is recommended and its free. “Digital Ocean” is another cloud technology that can be used by Linux. These can be used to store information or data in the cloud servers. They also allow different levels of security to help keep user information safe.  Linux also has built-in server tools inside the OS that can be utilized. The servers are among the most stable. The hardware is cheaper than IOS. | Out of all the available OS Windows has the most options for hardware and software. Windows is more user-friendly than Linux but not as easy to use as IOS. Windows also has the biggest support for developers. This will come in handy with setting up programs or with hardware.  Similarly, to the others, “Flask” can be used, and “React”. These programs will aid in the development of the application. Cloud-based technologies can also be used with Windows. The hardware is the cheapest of all three. However, this only applies to the early start-up. The hardware can start to get expensive if it's upgraded. | The application for mobile devices must be built on the backend by using react and flask. |
| Client Side | IOS hardware is on the more expensive side compared to the others. IOS is also generally easy to get started with considering the format of the OS and the functions.  “React” will be essential in building the application with IOS. This can be given to the developers, or it can be contracted out depending on the budget. This is another con with the React technology not all developers are experienced with it. | Linux may be the most cost-efficient OS out of the three. Most developers insist on using Linux because it was designed with programmers in mind. However, the con for this OS is that it is not as user-friendly as IOS or Windows. There will be significant time for developers to get acclimated to this OS if they are unfamiliar.  Just like IOS, “React” will be essential in building the application with Linux. This can be given to the developers, or it can be contracted out depending on the budget. This is another con with the React technology, not all developers are experienced with it. | Just like the others React, Flask, and VSCode will be helpful in building the application. Just like the server side, the support for Windows is extensive. The support is more than any other OS. On the client-side cloud technologies will be helpful as well. | Mobile phones are the most accessible hardware in the population. This allows the application to be accessed from anywhere a mobile device is. There must be some changes in the format of the application in order that it can be used on a smaller screen. React will aid in this problem because it will aid allow for the modification to the application. A con is that the developers who are doing this transfer must be familiar with this technology to accomplish this task. |
| Development Tools | An IDE that can be utilized in IOS will be mandatory, this will help with debugging and solve syntax issues. This tool also can help in connecting with other programs. A package manager will also be needed to install the different programs that will be needed to build the application. The IDE recommended is “Xcode” which will allow all the frameworks to be used. The package manager that is recommended is “Homebrew”. This will allow the installation of “Unix”, ”iTerm emulator”, “Dash API” and more when needed. | The developmental tools will be nearly identical to Linus as IOS. The IDE recommended is “Xcode” which will allow all the frameworks to be used. The package manager that is recommended is “Homebrew”. This will allow the installation of “Unix”, ”iTerm emulator”, “Dash API” and more when needed. The only real different will be the cloud bases technologies and systems that can be utilized. | The tools for development will be the same as the others. With addition of mySQL and google Chrome. | The application may need access to browser this can be a con because it will need to cross platform in all of them so that any user with any browser can access the application. Another consideration is that the application may need to go through the app store or the Google store to be downloaded onto the mobile device. |

