Oregon State University
College of Engineering

CS Capstone Technology Review

November 14, 2017

Ancestry Data Viewer

Prepared for

ADVR

Ashley McGrath ____________________ Signature ________ Date

Prepared by

Group 22
Ancestry Data Viewer

Le-Chuan Justin Chang ____________________ Signature ________ Date

Monica Sek ____________________ Signature ________ Date

Yong Ping Li ____________________ Signature ________ Date

Abstract

This document is describes the technologies needed to be used to create the viewer.

1 1.1 Overview

The User Interface (UI) is what the user will interface with in order to further examine and access additional features in the Ancestry Data Viewer. The UI should, as dictated by the requirements document, be both easy to use, and appear at least somewhat aesthetically pleasing.

2 1.2 Criteria

The criteria for the tool used is that is must be compatible with both Linux, specifically Ubuntu, and Windows 10. This is the most important criteria for the tool. Another important criteria is how well the tool can be used to create UI, since one of the criteria as referenced in the requirements document is that the UI must be aesthetically pleasing and easy to use. Another criteria is that the UI choice must be compatible with the tool or tools being used for displaying the family tree.

Criteria that is used in the event of equally suited tools will simply be to select whichever tool is easiest to use.

It should be noted that the primary component to fulfilling the usability and aesthetics component of the UI is more likely to be the result of design rather then the tool used to develop it, however, the selected tool is an important component for ease of development.

3 1.3 Potential Choices

4 1.3.1 JUCE

JUCE is a partially open-source library that uses OpenGL to generate a user interface. Since it runs on OpenGL, it is compatible with both Windows and Linux, and as an added bonus it is also compatible with Mac [1]. Another feature of JUCE is that it is also compatible with mobile devices [1]. JUCE is free for personal use, though a license is required for larger teams and larger applications. One notable source of JUCEs users are creators of music creation tools, such as FL Studios [2]. There are also various tutorials for using JUCE available on its website.

A helpful inclusion with JUCE is the Projucer, which is an application that helps with developing UI. By allowing the developer to manipulate the GUI by hand, be it resizing or changing the colors [3]. It can also allow HTML-like development, except that it allows C++. Projucer is also compatible with various IDEs, such as Visual Studios [3].

Another aspect of JUCE is that it supports plug-in development, especially for VST, AU and AAX [1]. It is frequently used for various plug-ins for various Digital Audio Workstation. Additionally, JUCE is compatible with mobile devices, both Android and iOS [4]. An aspect that is less direct but helpful and notable nonetheless is that there is a forum for developers, in case assistance is needed from experienced developers.

5 1.3.2 UMG UI Designer

UMG, short for Unreal Motion Graphics User Interface Designer, is a UI development tool for Unreal Engine [5]. It is a visual editor that allows for the creation of menus, and other interface graphics [5]. UMG is bundled and designed for use with Unreal Engine, so there is little need for an external UI library if Unreal is selected to be the exclusive visual development format.

UMG has an extensive tutorial available online, and it features an online forum to request assistance.

6 1.3.3 Unity UI

Unity includes a UI creator, which features a tutorial and forums to consult if needed [6]. The Unity UI creator is frequently used for various Unity games. There is a problem in that Unity UI is very resource intensive, to the extent that it runs twice as often as the default Update() function from Unity. The Unity UI creator is also a visual editor.

7 1.4 Discussion

For the purposes of developing the Ancestry Data Viewer, there is no glaring flaws with any of the three technologies. JUCE works with every application, while Unity and Unreal are optimized for their respective engines.

8 1.5 Conclusion

We have concluded that the tool used to create the UI is heavily dependent upon what application or applications are used to generate the image. For instance, if we are fully committed to Unreal Engine for every part of visualization, then it would be simple to simply use UMG, on the grounds that there would be no concern for compatibility, since it is built in. Alternatively, there is not a particular need to use it if Unreal Engine is not in use. Therefore, the UI framework would be Unity UI if the Unity Game Engine is used, UMG if it's Unreal, and JUCE if it's neither.

9 4 References

[1]https://juce.com/features

[2]https://juce.com/made-with-juce

[3]https://juce.com/projucer

[4]https://juce.com/maq

[5]https://docs.unrealengine.com/latest/INT/Engine/UMG/

[6]https://unity3d.com/learn/tutorials/s/user-interface-ui

[7]https://forum.unity.com/threads/ui-design-tool-pros-vs-cons-unitygui-guitextures-3d-mesh-or-3rd-party-tools.76937/