Anthony Medovar
CS 3744
Project 3

Summary Document

The purpose of this project was to create a mostly realistic prototype of our genealogy website. In this project, I used a Model View Controller (MVC) Architecture in order to present the web pages with maximum usability and functionality. Each webpage has its own unique look and function but there is a consistent style used throughout the website. The web app was designed to allow an individual to record their family history and create an online family tree that can provide relationship information and life events of each family member. The intended audience is anyone looking to keep a record of their family online. People love to be able to see their family on a tree and see all of the connections between those that they love. Simplicity is the theme for the website as well as providing a delicate human-computer interaction.

In order to have a functional website, there were many additions to this project that were not in the other project. First, the use of XAMPP allowed me to generate a personal server and database to house my files and data. Second, the use of PHP mixed into the already useful HTML, CSS, and JQuery turned the basic html pages from the project before to the usable webpages in this project. Another aspect that changed in this project was the use of a RESTful server and the ability to obtain and send information to the database through GET and POST requests to URLs. This project included much more tools and frameworks than I have ever used in the past.

With these tools, I was able to create many more pages on the website. These web pages include a home view, list view, detail view, add view, edit view, delete view, and a login page in order to access the majority of the website.

There are two different versions of the home page, an unregistered view and a registered view. This was done so a login page could be implemented and an account for the website could be made possible. The unregistered view is simple and provides general information for a user who may want to sign up for the website. The user is given the option to login. After the user logs in they are sent to the registered home view. This page contains personal data and links the rest of the webpages through the header, or navigation bar. These pages did not change from the original prototype other than the registered homepage getting its own navigation bar for a more consistent feel. The original mock ups were kept in mind throughout the design of these two pages.

The list view is the simplest view in the website. It contains a list of all the family members from the database. This is done with a GET request to obtain family member data and display it visually via a list. Other than the basic navigation functionality that most of the pages have, this web page provides a list of family members that are linked to their personal profile page. If the name of the family member is clicked, it will send the user to that individual's profile. This was not changed from the original idea of the web page. Nothing major changed in this part of the web page.

The profile page is also known as the detail view page. This page provides all of the information for a single family member and displays the information by using a GET request from the database. This information is generally easy to read. A detail page is technically a template page and is filled with specific family member data. This means that even if you have 100 family members, there is still only one detail page that fills up with each family member's information every time it loads. This page was also not changed too much from the original prototype. Mother and father data was put in to help with connecting the other family members.

The next three pages all have the same principle, to add, edit, or delete a family member in the database. The add view was necessary to occupy the database. This view was changed from the original prototype immensely in order to make it look cleaner and give functional ability for the user. The user

Anthony Medovar
CS 3744
Project 3
can fill out the webpage and relationships and press “Add New Family Member” and all of the data is sent to the database using POST request. The only required fields are first name and last name of the new family member. This data is stored into the database and is later obtained from a GET request. After the new family member is created, the user is sent to the new family members profile page. The edit view looks very similar to the detail view but allows for edits to happen. There are edit buttons located to change personal data, the photo, and to add life events to the individual. These edits use Post requests to send the updated version of the profile to the database and update the database using SQL queries. The Delete view is used to delete family members. This ca be done but looking at the list of family members to the right and locating the specific family members id to type into the input. Then, by clicking the “Delete Family Member” button, the user is sent to another page that will confirm the deletion of the individual. If the user clicks yes, the post request is sent to the server to delete the family member from the database. If the user clicks no, the user is sent back to the delete view. These three views create the general functionality of the website.

The last webpage that I created was the login page. This login page is very simple and straight forward, where all you need to do is type a username and password then click login in order to get to the full website. The login credentials for that I used was “**tony**” as the username, and “**gui**” as the password.

Setting up this project is simple for anyone. If someone was going to set up this project the first thing to do is open up XAMPP and start both the Apache and MySQL modules. After obtaining these files, place them into the “htdocs” folder in the xampp directory. From there, open up a web browser and type “http://localhost/cs3744/project3/tonym301” into the URL and you should be sent to the unregistered view of the homepage of the website.

Overall, I attempted to have minimal changes throughout the process of turning the high-fidelity prototypes into functional prototypes. I tried to adjust to a real world environment. This allowed me to find what I needed to keep and what I needed to leave out from the original html pages. This project allowed me to explore more frameworks and tools and gain an understanding of how real websites connect to servers and share data between web pages and databases.

**Above and beyond:**

There were many areas where I could have went above and beyond on this project. The minimal requirements and the ability to create anything that appealed to me, allowed me to explore web design on a whole other level. In order to get through the minimal requirements I needed to create the main views to have a functional website, demonstrate knowledge of POST and GET requests and how they are used for data management. I believe I created a website that did all of that and more. I supplied the user with the ability to create relationships between the family member as well as add life events. I included more JQueries in order to help with the functionalist of the webpage. I also created a consistent environment for the user. I put my best effort into making the webpages look appealing to any user. I worked on finding the best way to implement all of the webpages to have a streamline experience for the user. I believe I went above and beyond on this project.