George Eliot Archive Social Network Improvements

Developer Manual

Authors:

Tanner Finlay

David Braswell

Sponsors:

Dr. Beverly Rilett

Brad Hughes

Submitted in partial fulfillment of the requirements for COMP 4710 Senior Design to the Department of Computer Science and Software Engineering, Samuel Ginn College of Engineering, Auburn University

COMP 4710 Senior Design Project

*All sections of this report were composed collaboratively between Tanner Finlay and David Braswell

Table of Contents

1. Executive Summary

In this guide you will find important information related to the George Eliot Archive's Social Network feature and its operation. This social web visualizes George Eliot's relationships with friends and family members in this interactive social network display, sometimes called a relationship web or a personography.

https://georgeeliotarchive.org/relationship-visualization
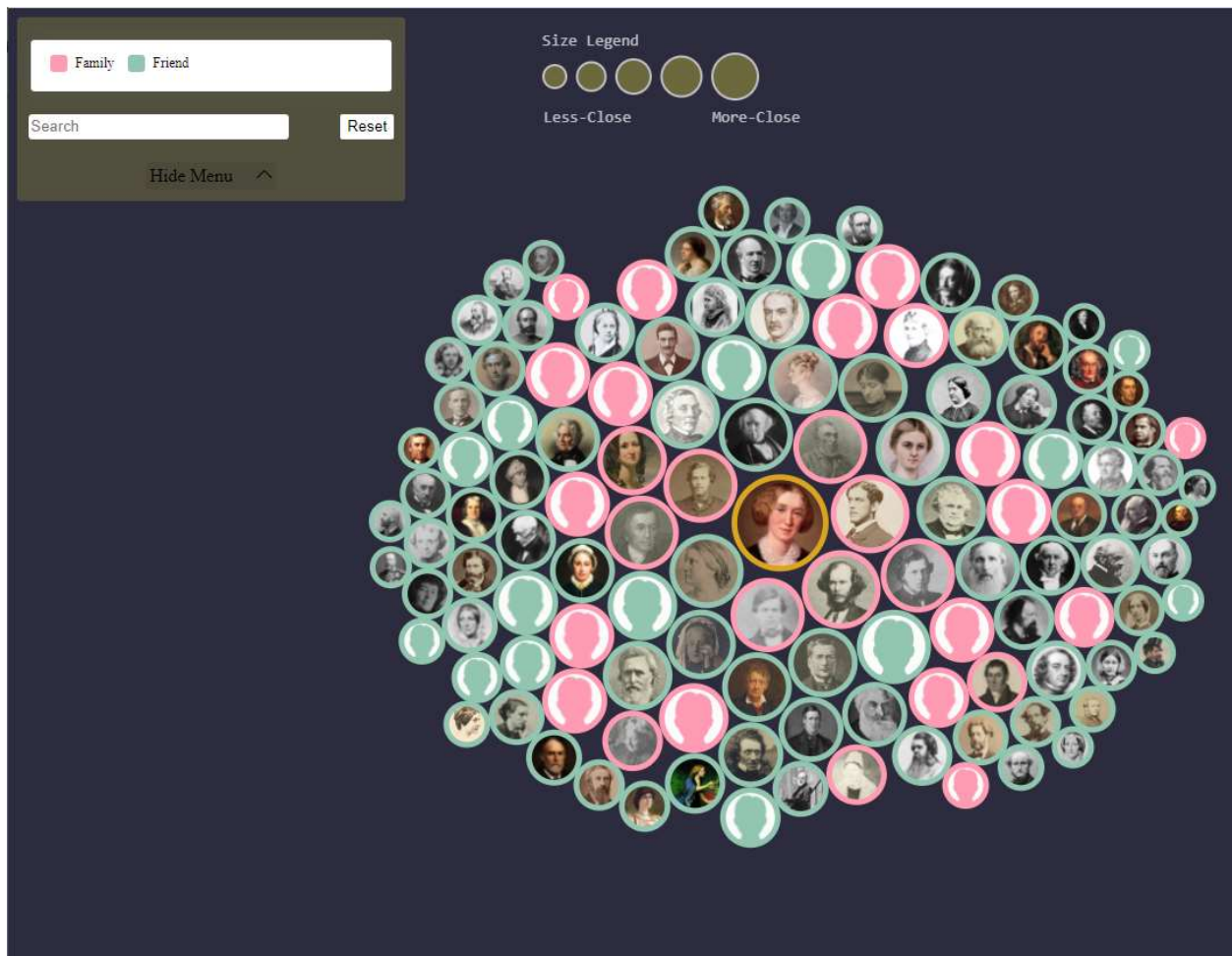
Notable project changes include:

1. Updated the relationship web dynamically with up-to-date information when you alter the item/description of a person in George Eliot's social network.

2. Migrated this map off GitHub and hosted it on the archive itself as a page component.
3. Linked to Wikipedia/biography pages for individuals referenced within the social web if applicable.
4. General bug fixes and improvements.

The purpose of this document is to inform future developers about how to navigate, maintain, and expand upon this web application.

2. Environment Setup

To begin setting up your environment, navigate to the github repository located at:

https://github.com/GeorgeEliotArchive/relationship

Fork the repo so you can push and view your changes as you develop. This is where you will be working. Make sure and create new branches and don't ever directly alter a file in the main branch. Use pull requests appropriately. When you have code that is ready to be released, see **5. Version Control**.

3. Future Work

The only user story we did not have time to complete was a redesign of the search function user interface.

UI Design Update	
Summary	As a user, I want a sleeker looking search bar in a more intuitive spot.
Description	The search bar currently is clunky to use and unappealing.
Planned Hours	7
Actual Hours	12
Programmers	Tanner Finlay, David Braswell
Testers	Tanner Finlay, David Braswell
Status	DNF due to time constraints

There is a partial solution in the code. Snippets can be seen below:

```css
/*--- SEARCH ---*/
.search {
    position: absolute;
    left: 50%;
    top: 50%;
    transform: translate(-50%, -50%);
    display: flex;
    gap: 2;
    align-items: center;
    border-radius: 65px;
    border: none;
    background: #68302C;
    padding: 30px;
    box-shadow: 0px 4px 4px 0px rgba(0, 0, 0, 0.25);
}

.submit {
    border: none;
    background: none;
    color: white;
    cursor: pointer;
    fontSize: 40px;
}

.search-txt {
    width: 0;
    border-radius: 65px;
    border: none;
    background: transparent;
    color: white;
    font-size: 40px;
    outline: none;
    transition: all 1s;
}

.active .search-txt {
    width: 350px;
    background: #F2F4F8;
    color: black;
    padding: 0 10px;
}
```
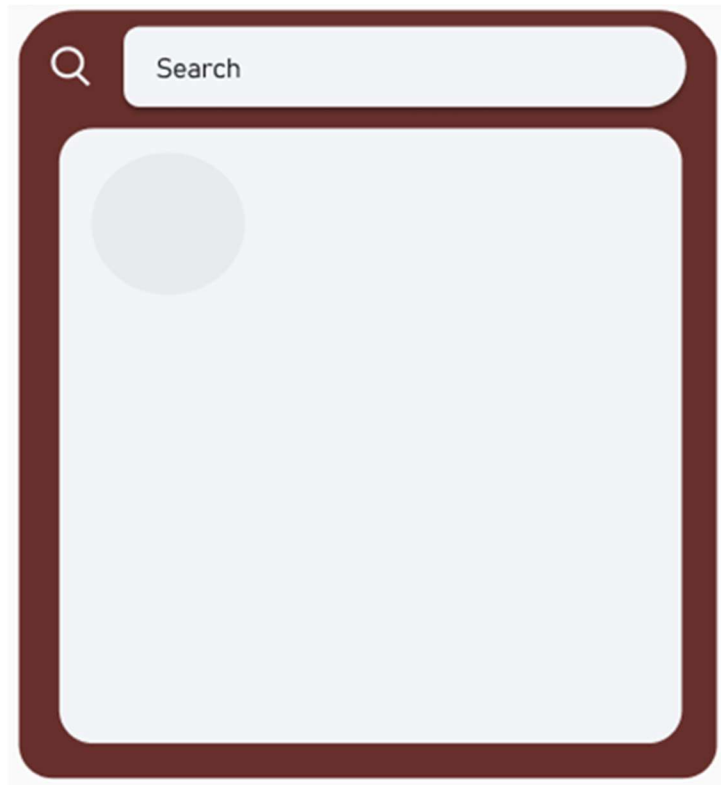
Our prototype was reflexive and animated to show as much of the social web as possible. This new configuration could be hidden with a click, and with one more it would expand again. It would animate one more time, expanding downwards as a person is selected and their biography shown.
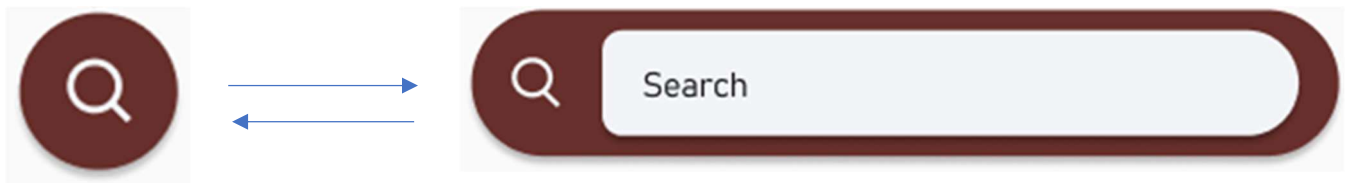
A prototype for this function was built, complete with input filtering and animation. Feel free to expand upon this design. Skeletal remains of html components, css, and an animation script can be found commented out in the repo.

4. Navigation

For those with administrative privileges, you will need to be able to edit information behind the scenes. To do this you will need access to the simple page on which the web is embedded and the backend cPanel file server. Below you will find instructions on how to navigate to both web locations.

From the admin home page:

https://georgeeliotarchive.org/admin/

Navigate to "Simple Pages" then locate "George Eliot's Social Network" and click edit. From here you can view all html code that references the web and the collection below. Images seen below the web were hard coded here with their file names as references.

To navigate to the file hosting backend, first login at

www.reclaimhosting.com

and navigate to cPanel.

Find "File Manager",

and then navigate to georgeeliotarchive.org:
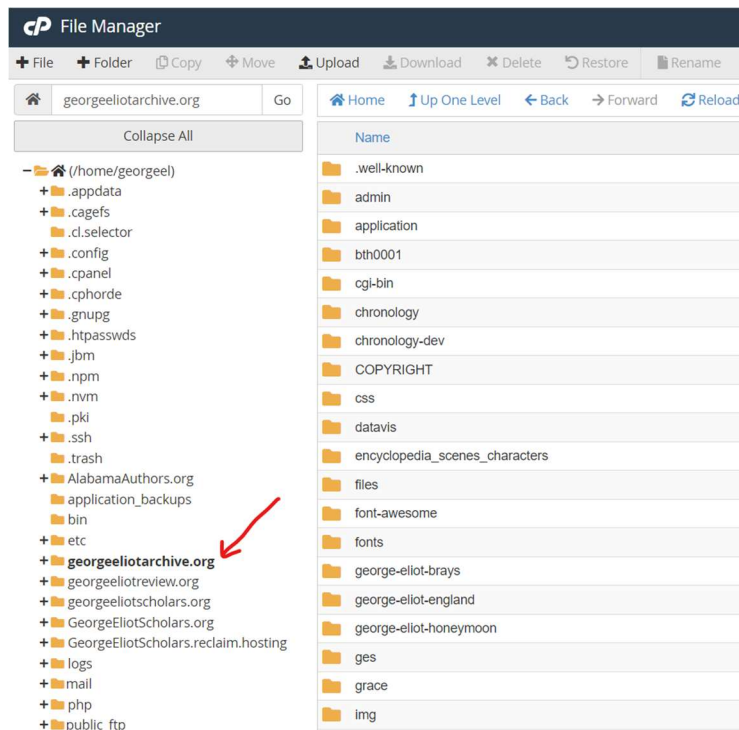
and finally, "relationship-web-updated." This is where the Social Network files are stored.
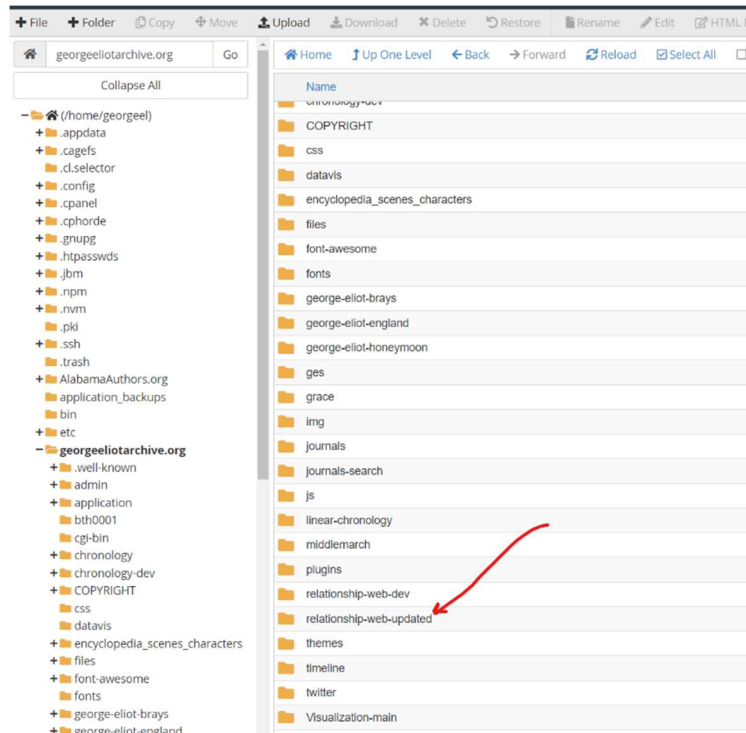
5. Version Control

When newer versions of the social network relationship web are developed and become available, there will be a need to update the version on Omeka. Fortunately, this is trivial.

To begin, navigate to the github repository located at:

https://github.com/GeorgeEliotArchive/relationship

Then, click "code" and "Download Zip":

Extract this zip file wherever you want, then navigate to the "relationship-web-updated" of the backend file server. See **2. Navigation** for details. Next, very carefully delete the contents of the "relationship-web-updated" folder and replace with those extracted from the zip file. WARNING: If you do this step incorrectly, the web will break and those files will not be able to be recovered. Use caution.

6. Updating Data

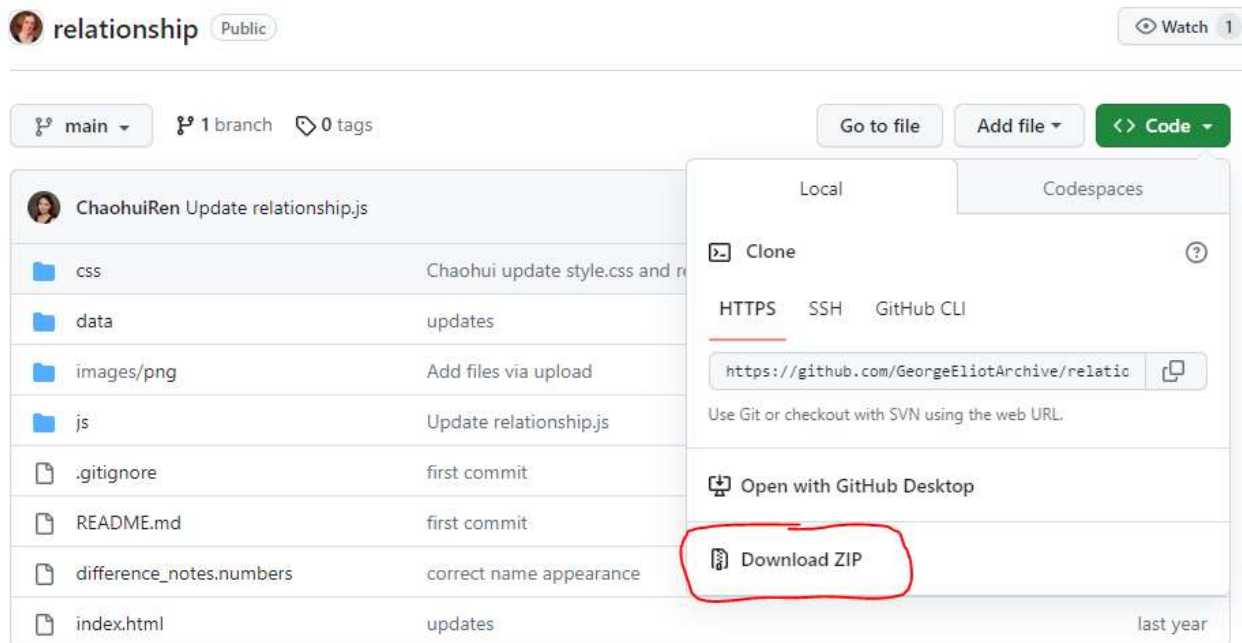The Social Network feature was created in JavaScript and uses the D3.js library. The data for the members is stored in multiple csv files and has entries for relevant information regarding the person's relationship to George Eliot. The archive uses a combination of JS, HTML, and PHP, with CSS for styling. No deep database altering is required since collections of .csv files are used.

JavaScript's Fetch API was used to download the Relationship Web Member Collection from Omeka's API endpoint. This gave us the information we needed in JSON format. Because we needed specific information from the JSON file, we made a script that retrieves the desired data and stores it into an array of objects that we then could write to a csv file. See dynamicData.js for more details.

Unfortunately, not all required data is able to be pulled from the collection API. Such information includes closeness, relevant links, image file names, gender, and relationship (family/friend). As it turns out, these values don't ever need to be changed. This presents us an opportunity to create a new csv, titled staticData.csv, filled with information that should not be updated with every API call. We then retrieved both spreadsheets separately and combined them into one data structure.

4.1 Replacing Images

To update images in the social web, the user will have to:

1. Download each image.
2. Find the "Image" column of the staticData.csv file and rename them exactly the same as the preexisting one.
3. Upload them to *relationship/images/png* in the repository, replacing the outdated ones.

The naming conventions are very important, as the code uses the "Image" column file names to reference which file to pull.

4.2 Replacing Static Data

The only data that needs to be manually updated is that in the staticData.csv file. This is as simple as editing the file and saving it back to the repository. Wiki links are to be pasted in the staticData csv under the "Links" column. Those with more than one link should have them delimited with a space and they will be added one at a time as html components.

7. Contact

For further information, please email Dr. Beverley Rilett at **bdr0032@auburn.edu** for all concerns.