

**Virginia Center for Civil War Studies eLibrary:  
Website redesign  
Course CS4624: Multimedia, Hypertext and Information Access  
Virginia Tech in Blacksburg**

Authors: Ben Sechrist, Suhas Avadhuta, Kevin Ellis, and Richard Howell

Client: Paul Quigley

Professor: Edward A. Fox

May 14th, 2015

# Table of Contents

<b>Executive Summary.....</b>	<b>2</b>
<b>User Manual.....</b>	<b>2</b>
<b>User Audience.....</b>	<b>3</b>
<b>User Experience.....</b>	<b>3</b>
<b>Developers Manual.....</b>	<b>4</b>
<b>Design Overview.....</b>	<b>4</b>
<b>Formats.....</b>	<b>4</b>
<b>Disqus.....</b>	<b>5</b>
<b>Delicious.....</b>	<b>6</b>
<b>Website Design.....</b>	<b>6</b>
<b>System Framework.....</b>	<b>6</b>
<b>Architecture Design.....</b>	<b>7</b>
<b>Design Prototype.....</b>	<b>8</b>
<b>Page Diagram.....</b>	<b>8</b>
<b>Database Architecture.....</b>	<b>9</b>
<b>Lessons Learned.....</b>	<b>9</b>
<b>Methodology.....</b>	<b>9</b>
<b>Timeline.....</b>	<b>10</b>
<b>Gantt Chart.....</b>	<b>10</b>
<b>Features list.....</b>	<b>11</b>
<b>Management Approach.....</b>	<b>12</b>
<b>Team Member Roles.....</b>	<b>13</b>
<b>Software Environments.....</b>	<b>14</b>
<b>Tools/Technologies Used.....</b>	<b>14</b>
<b>Prototype Stage.....</b>	<b>14</b>
<b>First Refinement Stage.....</b>	<b>17</b>
<b>Second Refinement Stage.....</b>	<b>19</b>
<b>Testing Stage.....</b>	<b>20</b>
<b>Future Work.....</b>	<b>25</b>
<b>Acknowledgements.....</b>	<b>25</b>
<b>References.....</b>	<b>26</b>

## **Executive Summary**

The purpose of taking up this project was to develop a website which would redesign and improve upon the existing <https://delicious.com/vccws> website. The function of this webpage will be to provide the Virginia Center for Civil War Studies an efficient way to organize and manage a listing of websites containing Civil War era sources. The improved website provides an more useful ways to access the many digitized sources on this period of U.S. history. Improving the site in these kinds of ways made it even more useful for the many students/historians/others who want to research Civil War topics but don't know where to start navigating the many content sites available on the internet. The site is a permanent feature of the Virginia Center for Civil War Studies website. It can be seen as the “go to” destination for anyone beginning research in a Civil War era topic, no matter how big or small.

We have four people on the development team. Although we make group decisions regarding specific functionality of various parts of the website, our client, Paul Quigley, takes an active role in making the big picture decisions. Our four team members are divided into two groups of two. Suhas and Ben have taken front-end developer roles, and focused their efforts on creating a usable user-interface, that interacts with the user well. The other two developers, Richard and Kevin, are developing back-end, server-side features that correspond to front-end UI features. Furthermore, we managed our project using the agile approach to software engineering. We work in one-week sprints to come up with prototypes, refine, and repeatedly test our development.

The website we have developed is designed to continually grow by increasing the number of Civil War sources that it offers as the user base expands. The admins of the webpage can manage all of the the users and the content that is presented on the main page of the website. Any of the users can recommend a bookmark to be placed on the website. Then the admin, like our client Paul Quigley, can decide to add or delete the requested source. This allows the website to be dynamic in nature and provide relevant content to the users. In this way, the website can become more and more useful as the number of users expand in the future.

## **Users manual**

### **Scenarios:**

For users trying to use this website for normal use, simply use the website in the same way that it was used in the past. Please see the future paragraphs and sections in this report covering the features we added, and how they are implemented/used.

### **Project Overview:**

Every year, more and more historical sources are digitized and made available online, from newspapers and magazines to handwritten diaries and letters. The Civil War era (1848—1877) is an especially popular period, and there are so many primary sources available online that it is difficult for historians—whether faculty members, students doing research projects, or members of the public—to keep track of what’s available. It can be difficult to know where to start when beginning a new research project. Therefore, the Virginia Center for Civil War Studies has created a listing of websites containing Civil War era sources. Our project is going to use this data and present it in an easy-to-navigate web application.

We aim to provide useful ways to access the many digitized sources on this period of U.S. history. The basis of the site is to continue to be a listing of the various content websites, accompanied by a short description and tags. We have made the website easy to navigate with a navigation bar and menus, and have also added new functions, such as allowing users to search. We can search the text of the short description, as well as the titles. We added the ability for users to add their own tags and comments. We provide a preview of the content websites on our site on the same page.

### **User Audience:**

The first users we looked at constitute the general public. This includes students, teachers, historians, and researchers. This user group will be mostly interested in a particular link, or a type of link, on our site. We will help them mostly by providing a description of each link and a search function to aid them in finding what they need quickly. The next group of users we have thought about, are developers. This includes the system admins, web developers, and database developers that will follow us in maintaining the site. Their biggest concern will be making sure the website continues to run smoothly and adding features they feel are necessary. To aid them in this, we have made sure our code is commented and clearly defined. The last group of users we need to think about are institutions. This group includes libraries and schools/universities. This group will be mainly focused on all the information we have to provide, and how they can get that information easily. To help this user group, descriptions will not be shown at all points and times, only when the user asks for them to be shown. This makes it easier for groups trying to get all the links at once without having to scroll through unnecessary information.

Developers: System Administrators, Web Developers, Database Developers  
Institutions: Schools, Libraries, Universities

### **User Experience:**

When we built our system, user experience was one of our top priorities. We wanted to create a website that is simple to navigate. In order to do this, we provided a user interface which is familiar to the users. Our main page of the website contains links clearly labeled with concise

descriptions of what kind of information the link holds. They were going to be presented in an accordion view (feature was reversed in final product) so there is less clutter and the user can focus on the titles instead of all the small details. Once the user has chosen which link he prefers to investigate further, he can click on it to expand the accordion and reveal a preview of the page. This design will allow the user to discover the resources they need quickly and efficiently. If you were a previous user of the Delicious website then you will have no problem navigating this new website as it includes some of the same features of the Delicious site and more.

## **Developers Manual**

### **Design Overview:**

For the design of this website, we use several existing libraries of Javascript and CSS styling. We are utilizing some popular, well-known frameworks such as 'Bootstrap' (for CSS and the way the overall website looks), Font Awesome [6] for icons that are easy to manipulate, and some AngularJS libraries that allow for some pretty robust features out-of-the-box. Combining these technologies has allowed us to create a nice-looking, functional website, with features the previous version did not have (e.g., Searching amongst links, Users, Favorites). For database management and server side operations, we will be using a MySQL database and PHP. PHP will allow us to communicate with the MySQL database easily and will help create and maintain user logins and profiles. The MySQL database will allow us to store all the information the site needs to keep track of, including users and links.

### **Tags and Bookmark Site Formats:**

When discussing the requirements with our client, Professor Paul Quigley, he informed us that they currently use Delicious' tag system for grouping relevant bookmarks together and for describing the format of the content on the site bookmarked. The way we are solving this is described later when the database modeling of tags is discussed. To represent the tags on the front end we use a type of word cloud similar to Figure 1. When a word is clicked the associated bookmarks with that tag are filtered and shown in the place of all the tags with an animation. The filtered tag is then shown above this list near the search bar.



Figure 1: An example of a word cloud.

### Disqus:

One requirement was that any user of the site, registered user or not, should be able to comment on bookmarks. To accomplish this we use Disqus [7]. Disqus allows for easy moderation and setup of comments that can be injected onto all of the individual bookmark pages. It also handles authenticating users and allows users to login using a variety of platforms or by registering with their email. Our client, Paul Quigley, can then be alerted when someone would like to comment, and can approve or deny it. Disqus also allows for our client to change settings at any time without the need to change code as it is independent of the site code.



Figure 2: Example of Disqus

## **Delicious:**

We initially need to get the existing links that the current website uses from a feed hosted by the Delicious service. Then we can manipulate them in our own database and add any necessary qualifiers to provide additional functionality.

## **Website Design:**

The website's basic functionality is to host a list of links (bookmarks) that are relevant to the Civil War in some way. Each bookmark has an associated short description that a user can use to decide where to go for the information desired. The website will have two more primary pieces of functionality (we are working on adding more features if there is time). These are a search ability and a preview window.

The search function will allow users to search through the links with keywords that matter to the user. Matches are refreshed instantly with no need to hit 'enter' or refresh the page. (The Angular JS libraries allow for some powerful, useful things!) The preview window will allow the webpage being linked to by the bookmark to show up iframed-in on the current page, without forcing the user to lose their place in the current list.

We opted to use a small list, which, when clicked on, expands to show details of the particular bookmark selected. Our color scheme is also similar in style to Twitter (because Twitter created bootstrap). However, our client has asked that we change the color scheme to appear more like the Virginia Center for Civil War Studies site [8].

## **System Framework:**

Our project will contain three main layers: User Interface Layer, Logical Layer, and Database Layer. These layers will allow us to keep separate the various functions of the system and organize the code base in a meaningful way.

### *User Interface Layer:*

This layer contains all the front end design elements that the user will see. The majority of this code will be in HTML, CSS, and Javascript. For example, the design of the various tabs, listing of the bookmarks, drop down menus, and layout of the various sections of the page will be developed in this layer.

### *Logical Layer:*

The logical layer acts as a bridge that connects the User Interface and the Database. It is responsible for retrieving and updating the data from the database. For example, if the user creates an account on the webpage, the logical layer is responsible for transferring the account information to the database.

### *Database Layer:*

This layer is the database. It stores all the information regarding the users, bookmark data, tags, and so on. The logical layer uses this information to update the user interface.

### **Architecture Design:**

We plan on using a hybrid of the client server and data centered architecture styles. With the combination of these two styles, we will be able to organize our system in a way that will satisfy the functional and nonfunctional requirements of the system. The client-server architecture will allow us to distribute our services to a wide audience. It will allow multiple users to access our product from a range of locations. The main reason we needed this type of architecture is because we rely on a system of networks with servers to store and provide web services as well as clients to use these services.

Our growing database of Civil War Resources will be shared by the clients, and because the servers can be replicated, they may also be used when the load on a system is variable. For example, the majority of our nonfunctional requirements depend on multiple users wanting fast web interaction from wherever they chose. For example, we plan on having a mobile application of our product. The client server architecture will allow us to implement this feature without change in our design plan since the user can be anywhere to interact with our shared database. In addition, we plan on using VT servers to support the clients. This will allow us to manage server maintenance and feature upgrades efficiently.

On the other hand, the data centric architecture is exactly the type of style we need to manage our large collection of bookmarks links, tags, users, etc. Since our nonfunctional requirements demand that we have quick searching of our various sources, the data centric architecture is perfect for this purpose. Additionally, we need to store the large volume of information in our database for a long time for continuous use by the clients. The data centric style will allow our essential information to be managed in a central repository that is accessible to all system components. This is exactly how access to the database should be set up since we don't want direct interaction of the components. All information must be gathered from the repository itself. Although the repository may be a single point of failure in this architectural style, there are considerable advantages for us in the way we manage our database and integrate it into our system when future changes are made.

Figure 3 below is a page diagram that shows how the sites will interact with one another.



## Design Prototype:

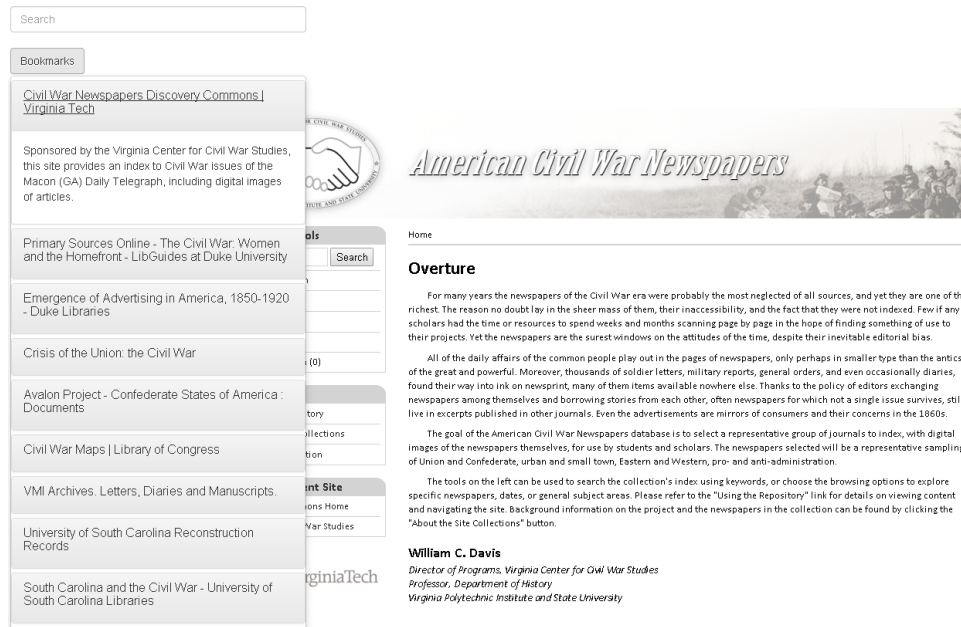


Figure 3: A initial prototype of our main page.

As you can see from this initial prototype (Figure 3) that we created, the design of the webpage will be very intuitive and simple for people of all ages to use. This is just an initial mockup of the webpage. There will be many iterative design changes on top of this design. However, some definitive design elements that are fundamental to our system will be an efficient and live update search bar, a compiled list of Civil War sources with descriptions, and a preview of the web page available to the user when they click on the link.

## Page Diagram:

Our site infrastructure will look something close to Figure 4.

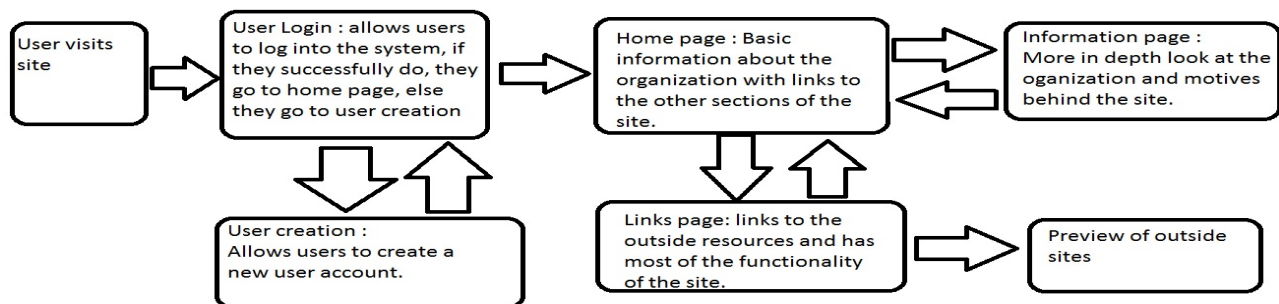


Figure 4: A page diagram of the site.

The arrows in the above description mean that a user can go from their current page to that page. This creates a clear and concise way for a user to use our site.

**Database Architecture:**

We are using phpMyAdmin as our SQL server manager. We have created new tables to hold data for the bookmarks, as well as users (new feature!) and their respective passwords. We are using a PHP encryption method called password\_hash() that uses a bcrypt algorithm to generate a random mapping. This keeps the passwords protected, even if the database is compromised. The PHP method used also generates a strong salt for us, so the security of the passwords are strongly enforced.

Figure 5 is an example of the user database

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)			No	None	AUTO_INCREMENT	Change Drop Primary Unique More
2	username	varchar(50)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique More
3	password	varchar(255)	latin1_swedish_ci		Yes	NULL		Change Drop Primary Unique More
4	timecreated	timestamp			Yes	CURRENT_TIMESTAMP		Change Drop Primary Unique More

Figure 5 : An example of a database to hold user information.

**Lessons Learned**

**Methodology:**

We planned to use the agile/scrum [9] methodology to gather, analyze, and validate the requirements. The requirements gathering was done with active stakeholder participation as well as group collaboration. For the first release cycle we focused on the fundamental and core features of our system. For each feature to be developed in the first release cycle, we gathered the details in the form of user stories. While identifying each feature, we made sure to utilize natural language while describing them. After prioritizing and estimating the features, we planned to release a new feature every week.

Using the Agile method to estimate everything is simple and relative. Each feature is assigned numbered estimates that have a unit-less measure. We know that features are the largest scope and under them are stories that are implemented with tasks but the estimations are all unit-less. For our estimation we used a scale of 1, 2, 3, 4, and 5. 1 is the lower/lesser end of estimation and 5 is the high end of estimation. We used this scale for features and stories, and because this is an agile approach the simplicity of agile means that these estimations include analysis, development and testing all in one number.

## Timeline:

Task Name	Start Date	End Date	Duration	Predecessors	% Complete
<b>Planning Phase</b>	<b>01/25/15</b>	<b>03/04/15</b>	<b>29</b>		<b>100%</b>
Requirements Gathering	01/25/15	02/05/15	10		100%
Learning New Technologies	02/10/15	02/23/15	10		100%
Designing	02/20/15	03/04/15	9		100%
<b>Prototyping</b>	<b>02/10/15</b>	<b>03/26/15</b>	<b>33</b>		<b>37%</b>
Setting Up Dev Environments	02/10/15	02/18/15	7		100%
Back End Development	02/18/15	03/25/15	26		30%
Front End Development	02/21/15	03/25/15	24		30%
Plan For Refinement	03/24/15	03/26/15	3		0%
<b>Refinement</b>	<b>03/26/15</b>	<b>04/15/15</b>	<b>15</b>		<b>0%</b>
Testing/Bug Fixing	03/26/15	04/15/15	15		0%
Possible Design Changes	03/29/15	04/15/15	14		0%
<b>Testing</b>	<b>04/15/15</b>	<b>05/05/15</b>	<b>15</b>		<b>0%</b>
Ad-Hoc Testing	04/15/15	04/16/15	2		0%
Thorough User Base Testing	04/16/15	04/29/15	10		0%
Bug Fixing	04/29/15	05/05/15	5		0%

Figure 10: A data view of the Gantt chart.

## Gantt Chart:

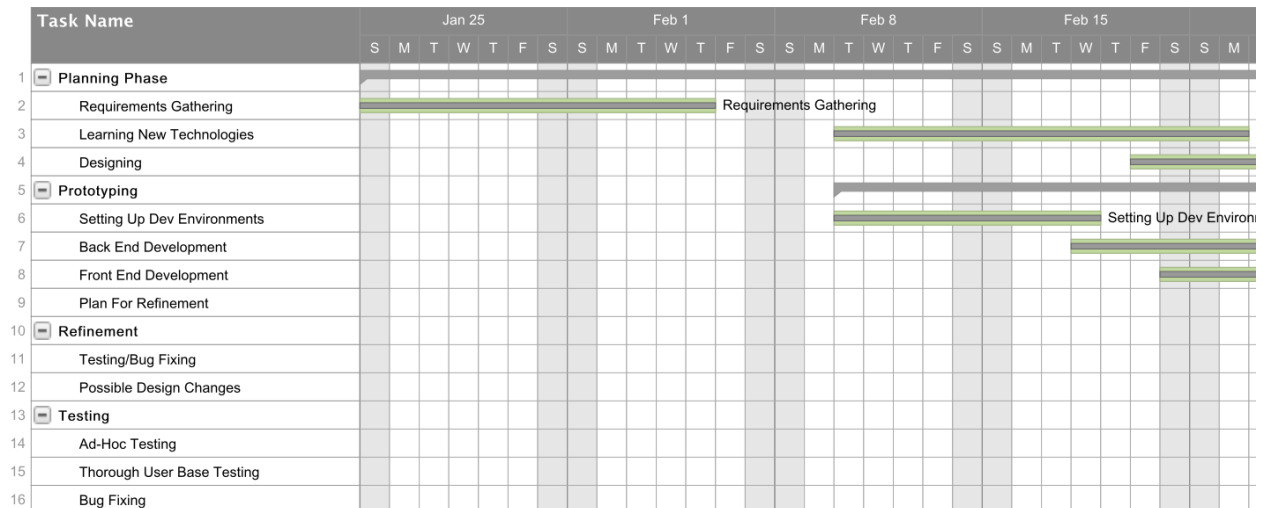


Figure 7: The left third of the Gantt chart.

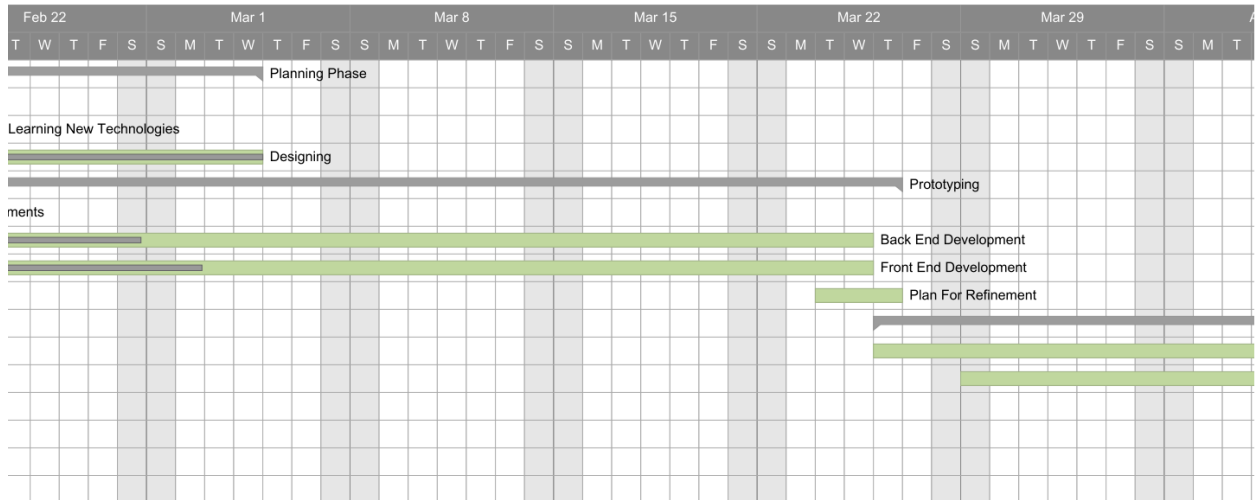


Figure 8: The middle third of the Gantt chart.

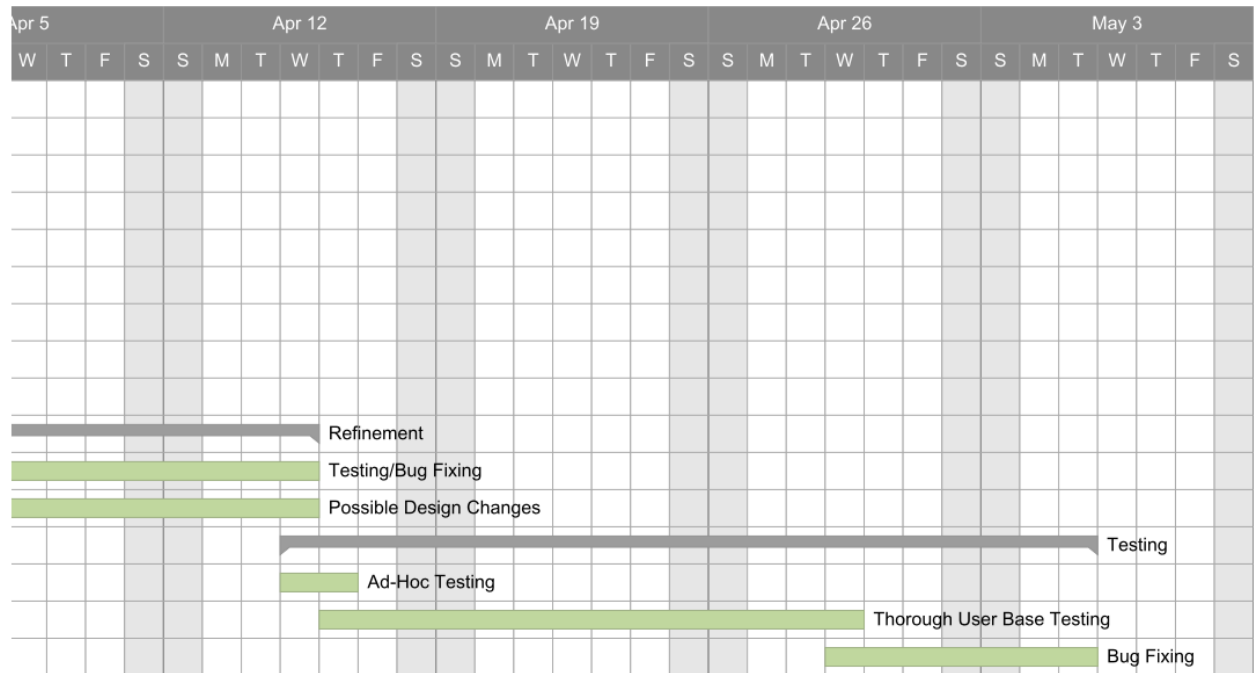


Figure 9: The right third of the Gantt chart.

### Features List (Product Backlog):

We planned to use include in our product a set of features to better aid in the client’s needs. The list below shows the features we initially planned on including, divided up into categories.

Feature 01: Set up Web server

    Story 1: Set up WAMP (Windows Apache MySQL PHP)

    Story 2: Set up backend resources

Feature 02: Set up Database

- Story 1: Create SQL Database
- Story 2: Load links and data from delicious site
- Feature 03: Prototype Home Page
  - Search
  - Nav Bar
  - Accordion view for Links
  - Styling and Design
- Feature 04: User tools
  - Tagging
  - Commenting
- Feature 05: User Accounts
  - Usernames
  - Passwords
  - Backend storing of info
- Feature 06: Log In Page
  - Design
  - Authentication system

**Management approach:**

As with any group project, there needs to be some central authority for task assigning and questions regarding steps. For this project, we are using project management software called trello. This allows us to have a place for our team to go, and update the rest of the team on progress. This is a piece of software that follows the agile approach to software engineering. We work in one-week sprints, and gather cards from a 'backlog' that have been created previously. When a developer needs to ask a question to another dev, they can add a comment to a card and tag others in the comment, which will automatically notify that person that they have someone asking them a question. There is no specific 'project manager' as we all are equal in design choice, but we do ask our client constantly for guidance when we don't know where to go next.

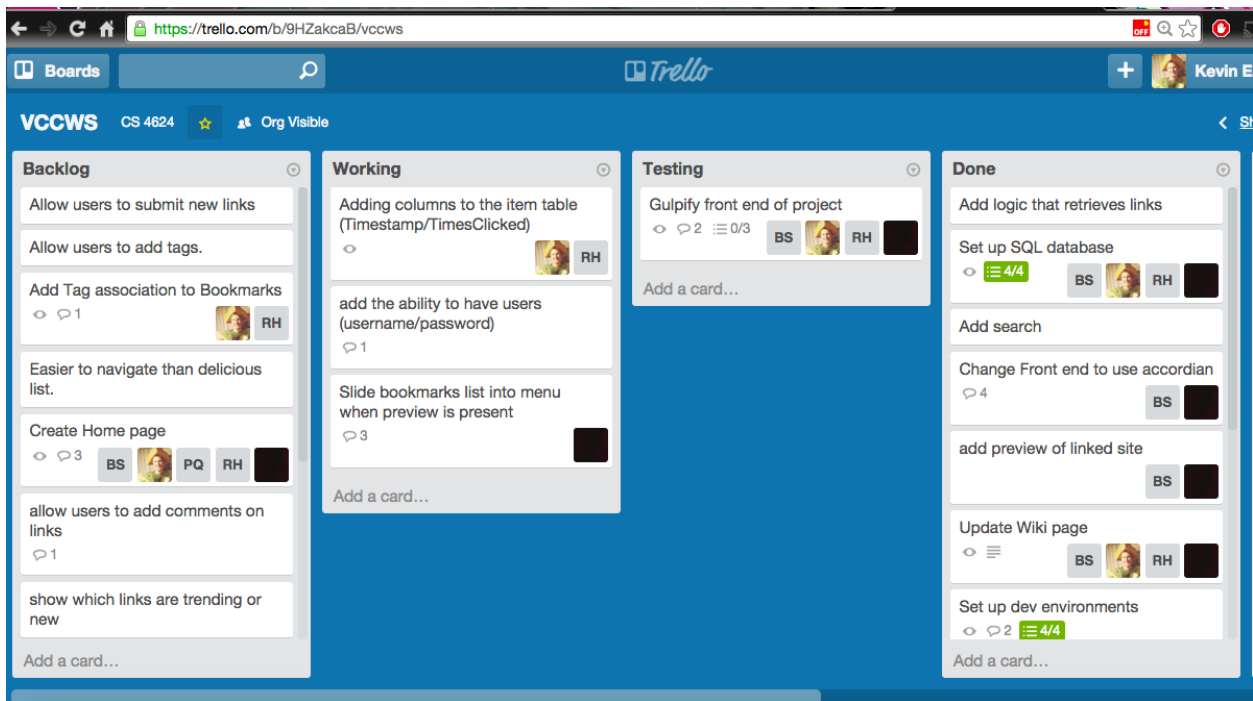


Figure 6: The trello page for the development of the site.

### Team Member Roles:

We have four people on the development team, and one client acting as a B.A. for decision-making purposes. Paul Quiggly is the client for the Virginia Center of Civil War Studies and takes an active role in decision making for our stories on a weekly basis, we meet with him every thursday morning.

Our four team members are divided into two groups of two. Suhas and Ben have taken front-end developer roles, and focused their efforts on creating a usable user-interface, that interacts with the user well. The other two developers, Richard and Kevin, are developing back-end, server-side features that correspond to front-end UI features. These stories center around database management, and allow our website to store information about users, or bookmarks or any other data we need to remember. All of the developers have agreed to these roles, and take an active approach to managing the project via the trello website.

### Software Environments:

For this project, we have decided to use AngularJS as a javascript framework, and Netbeans as an IDE (Integrated Development Environment). We write our code in that environment, then use git as version control. We have our project hosted in a Gitlab repository, where we can all pull/push to. This gives us accurate, up-to-date code as soon as someone edits a file. We are using WAMP server to allow the website to be easily hosted on our dev machines.

## **Tools/Technologies Used:**

**Web Hosting Server:** Web hosting allows us to make our website accessible via the World Wide Web; therefore this technology is fundamental for the success of our project. We will be using a third party web hosting service called WAMP [1] to support our project. Although this is how we will be starting our project, we will be looking to move our system to the Virginia Tech hosting service in the future when the web application is more stable.

**Styling and Design:** We are using Bootstrap [2] for CSS styling of all of our web elements. We are also plan to use a few Bootstrap elements such as modals and accordions. Bootstrap is also helping us to keep the web site very responsive and mobile friendly.

**JavaScript:** Most, if not all, of our JavaScript will be using the AngularJS [3] framework. This allows us to work with our data models in a very easy way. It also allows for dynamic content and a single page application.

**PHP/MySQL:** We chose PHP to communicate with the MySQL database and to help facilitate the creation and use of users. With PHP, we are able to store information between pages, and are able to have users give us information to store. We chose MySQL as our database system because of how easy it is to communicate with through PHP. With the PHP/SQL combination, information can be stored, gathered, and manipulated easily.

**NetBeans:** This is our IDE of choice. It allows for easy website deployment to our local environment and testing on that environment. It is how we will write the PHP code for the application.

**Trello[4]:** As an agile scrum board, this software allows us to monitor tasks that each team member is currently working on, has worked on, and will work on, all in one place. It also gives us a place to discuss the challenges, details and other things relating to specific features of the website.

**GitLab[5]:** We are using Git for version control, and the website GitLab to merge branch changes into the master branch. This keeps all our code up to date with each other and

**HTML5:** We are using HTML and its current features to our advantage with certain CSS libraries. This gives us complete creative control over the appearance of the website.

## **Prototyping:**

The prototype has some great initial design and functionality, but there are some changes we are planning in the final version:

- Collapsible menu on the left side of the screen

- We would like to save some screen real-estate and have the menu with all the bookmarks be collapsable to the side of the screen
- The purpose of the collapsable menu will be to give the user a larger area to preview the web pages while they are searching through the list of sources in the side menu. By sliding the menu to the left, the page will clear up and the preview will be larger.
- The user will be able to search from the menu itself to filter down on the sources they are looking for.
- Priority: We are definitely planning to implement this change because the team sees it as a fundamental design change which will greatly improve user experience and aesthetics of the page.
- Here is how the menu looks currently:

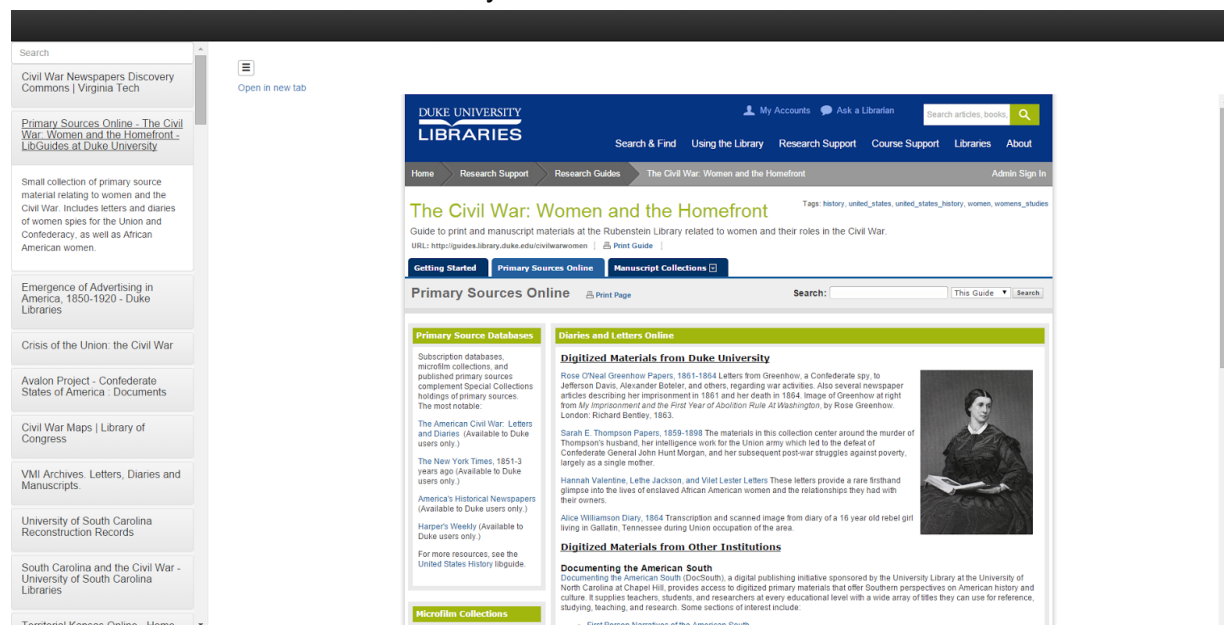
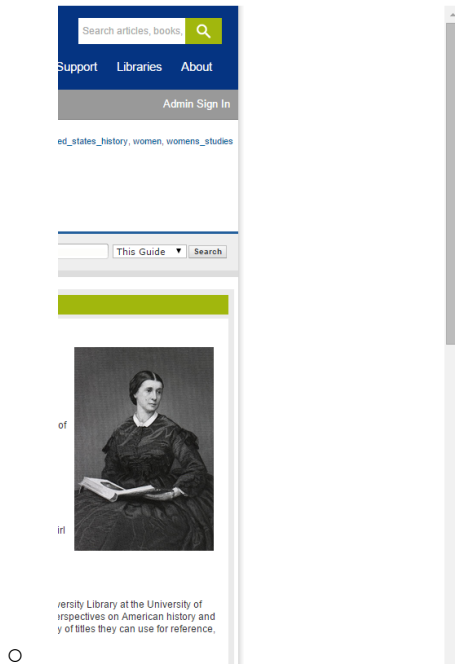


Figure 11: A screenshot showing the collapsable menu.

- Adding tags to the right side of the screen
  - We have some space to use on the right side of the screen where we are going to place tags that users can use to narrow down their search for specific bookmarks.
  - The tags will allow the user a quick and easy way to navigate the numerous sources which are listed in the menu. They will always be readily available to the user so they are easy to discover and use.
  - Priority: The space on the right of the screen will definitely be used to improve the design of the web page. Whether it will be tags or something else is still under consideration due to possible time constraints. However, we are hoping some sort of section will go there to make user interface more efficient.
  - This area is reserved for adding the tags:





*Figure 12: Screenshot showing where the tags will go.*

- Favorites
  - We would like for users to be able to favorite sources and keep them on their profile.
  - The user should be able to favorite the sources by clicking on a small icon which will appear next to the links.
  - The favorites bookmarks will allow the users to quickly see a list of their favorites and the ability to navigate to that particular source.
  - The favorites list will be attached to the users account so it will always be available to them when they log into their accounts.
  - Priority: The creation of favorites will be linked to the creation of the login page, which extends user accounts. The back end and front end of user accounts and account management have to be dealt with before favorites can be created. The development of all these aspects will definitely take up a significant portion of the project time so for now it is our stretch goal to complete this.
- Home/Login page
  - We would like to include some form of login where a user can save some 'favorite' sites this will use many to many relationships in our database.
  - The login page will be easy to use, and simple to navigate. We definitely want the users to continue to the main page without forcing them to create an account or login.
  - Priority: The creation of the login page will depend on the completion of the user accounts feature. The back end and front end of user accounts and

account management have to be dealt with. Since these tasks will take a longer than other planned features to complete, it will be one of tasks we will do later on our timeline. The login page is not fundamental to the redesign so it will possibly be a stretch goal to complete this.

## Refinement I:

During the refinement stage, there weren't any major changes we intended to make to the prototype because our client really seemed to like the design and placement of all the web elements. However, there were some minor changes we did to improve the aesthetics of the webpage and usability of certain features.

- Collapsible menu
  - After the collapsible menu was developed, we realized that it was blocking the search bar when the menu was open. So, we incorporated the search menu into the menu so it would also collapse in and out with the rest of the bookmarks. This made sense because the users were using the search bar to search for specific bookmarks.
  - The size of the search bar was readjusted to look appropriate to the webpage
  - The menu toggle button was adjusted for aesthetic purposes.

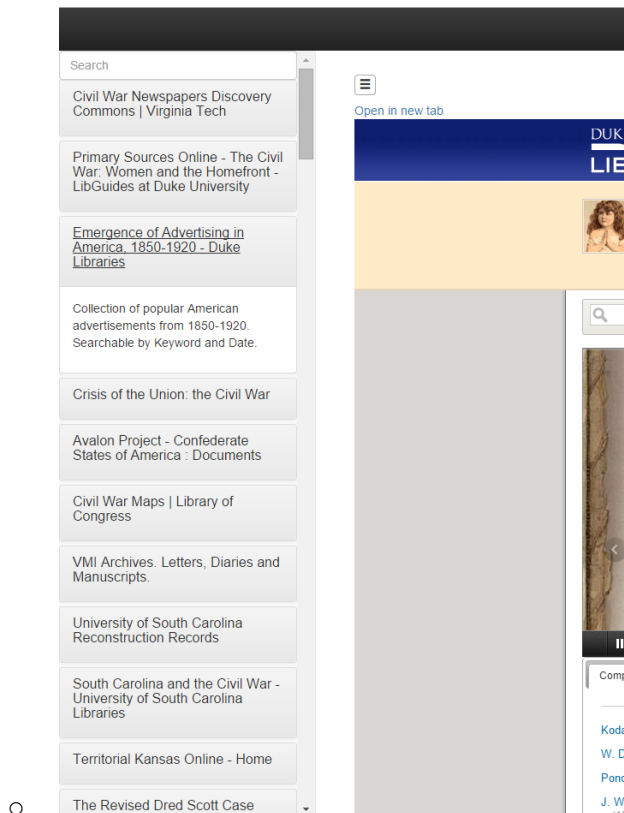


Figure 13: Screenshot showing the collapsible menu improvement.

- Home page
  - We intend on having a homepage with a link to an existing Civil War website.
  - The style of the header will use the existing style of the existing website.[8]
  - This page is not intended to have any real functionality, but rather to make it seem inviting
  - We received this information from the client:
 

“I think something pretty brief that contains a quick description of the project. Something like: "Explore the wealth of digitized primary sources on Civil War era history. Browse our listing of the best databases online. Click on the tags to explore by theme. Or search for whatever interests you. Whether you're beginning a major research project or just looking for fascinating new insights into the Civil War era, you're in the right place." I think the home page will probably be most effective if it is pretty clean, with an inviting and easy to find search tab; a set of tags (maybe in word cloud form) to click on; and the first part of the listing of sites for people to browse. (Maybe then you scroll down to see more of the list). Then we should have a link to a page called something like "About the project" or "About online research" - and have an updated version of the following page there.<http://www.civilwar.vt.edu/wordpress/gateway-to-digitized-sources-in-civil-war-era-history/>”
  - Header will look like this:



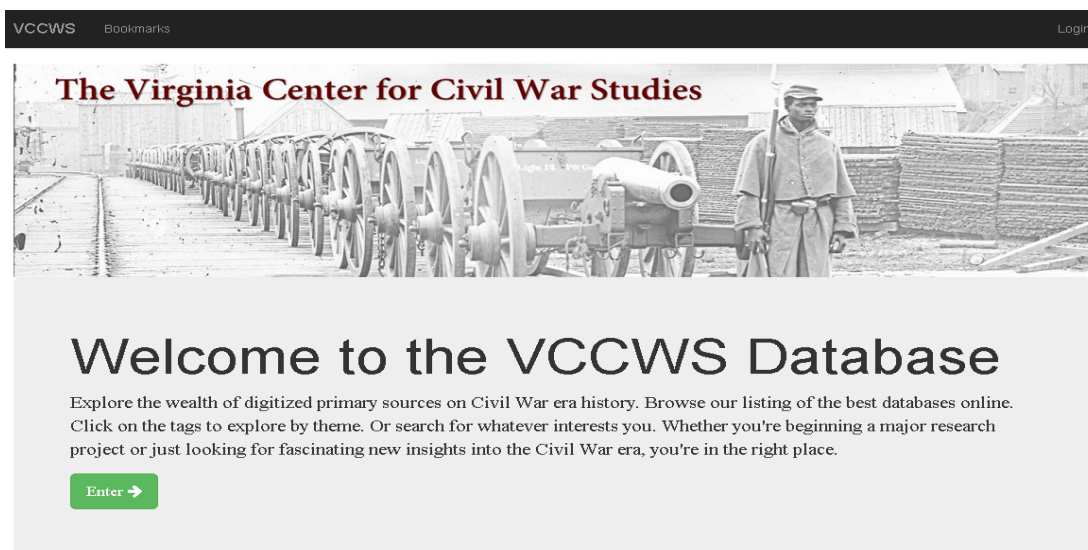
*Figure 14: The Header image for the site.*

- Admin Controls
  - Admins will be able to review link suggestions that users have posted for them and decide which should be added to the site.
  - Admins will also be able to promote users to admins, or remove users as admins, as they see fit.
  - Only users that are currently admins will have these options.

## Refinement 2:

During this second refinement stage, we really focused on improving the home page to be simple with a modern design. Additionally, we focused on including user authentication so that users can log in to save changes such as keeping a list of their favorite bookmarks. However, we don't require all the users to make an account with us. Additionally, there were minor bug fixes with some UI elements.

- Home page
  - One of the major features that is now fully implemented and refined is the home page. The home page is meant to give the users an idea of the type of content our website holds. It was designed to be simple and elegant giving access to all users without having to login in or make an account.
  - We refined the home page to give it the look and feel that our client likes. Our client provided us with a banner image and also some text which are both showcased on the home page. See Figure 15 below.



*Figure 15: A screenshot of the homepage.*

- User Login
  - Once logged in, users can suggest bookmarks.
  - Users can be promoted to admins by another admin.
  - Admins are considered users with additional permissions and actions.

- Admins can suggest bookmarks, approve suggested bookmarks, and delete/promote/demote users.
- Here is a screenshot of the login:

Login

---

Username

Password

---

Login Cancel

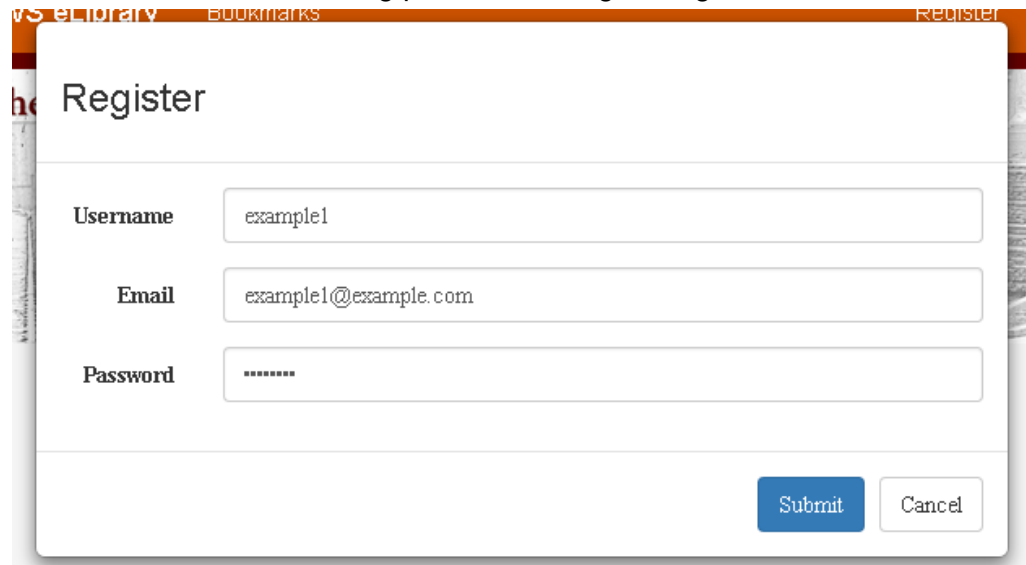
- *Figure 16: A screenshot of the login window.*

### Testing:

During the testing stage we were really sought out to do a thorough usability and discoverability test to see where we could make improvements from the users point of view. Since the focus of this project is mainly on the redesign of the previous Civil War Studies website, we also asked our client repeatedly for feedback on how he would like to see the webpage. There wasn't a significant amount of testing for the logical aspect of the webpage since the purpose of the webpage is to provide resources for the user in a clean and efficient manner. However there were some edge case bugs we found as we were doing the project. For example, there were some bugs in the back end related to registering users, and also when logging in. We have also been finding and fixing design and formatting bugs. All of the bugs anyone finds are being communicated to all of the group members through the use of an on going group messaging system. Any major bugs have been logged as tasks using Trello. The following show some examples of how we conducted some tests:

- Registering Users
  - Testing for this aspect of the webpage consisted of creating a new user and then making sure that the user can login afterwards. While running through any exploratory test, all steps for each process was carefully observed for any

bugs. From the clicking of the button to the expected output. The following screenshots show the testing process for Registering users:

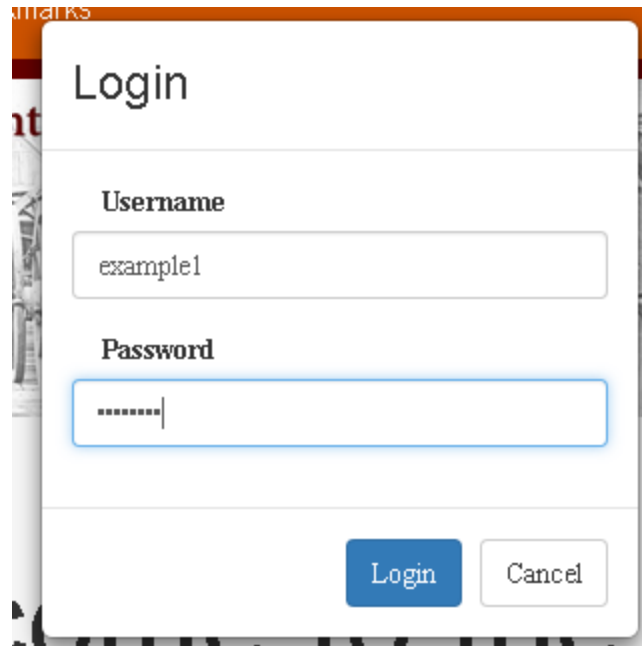


The screenshot shows a web browser window with a modal dialog titled "Register". The dialog has three input fields: "Username" with the value "example1", "Email" with the value "example1@example.com", and "Password" with a masked value of seven asterisks. At the bottom right of the dialog are two buttons: a blue "Submit" button and a white "Cancel" button with a grey border.

○

*Figure 17 : Registering users*

○ After we register, we want to make sure that we can login

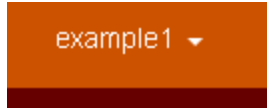


The screenshot shows a web browser window with a modal dialog titled "Login". The dialog has two input fields: "Username" with the value "example1" and "Password" with a masked value of seven asterisks. At the bottom right of the dialog are two buttons: a blue "Login" button and a white "Cancel" button with a grey border.

○

*Figure 18 : Login window*

○ After we login, we want to make sure that the username appears on the top corner.



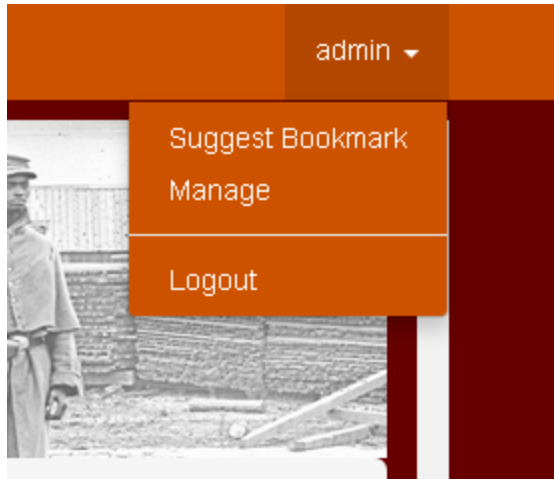
○

*Figure 19 : Name of user on top corner of page*

○ This is how you can verify the process of registering and logging in as a user.

- Admin Users

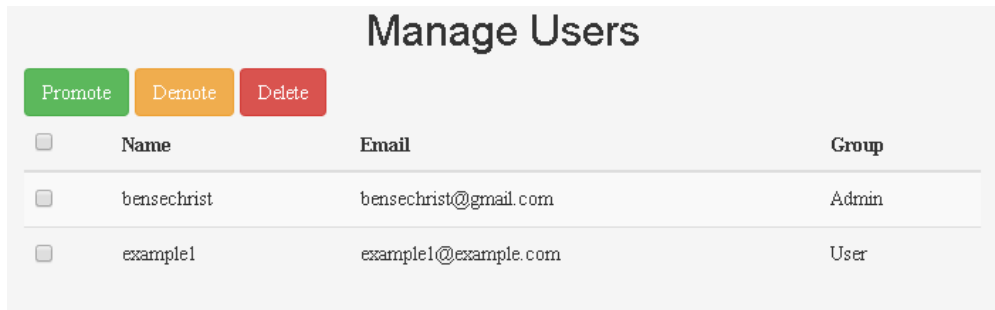
- Additionally, admin users should be able to manage users.



○

*Figure 20: Dropdown for admin users*

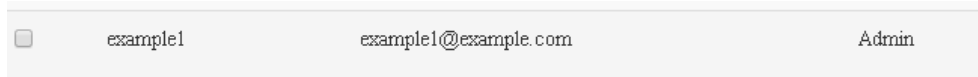
- As you can see here, the functionality can be tested by pressing the buttons to promote, demote, or delete users.



○

*Figure 21: Managing users page*

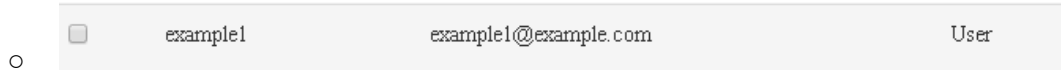
- Testing to see if example1 is promoted:



○

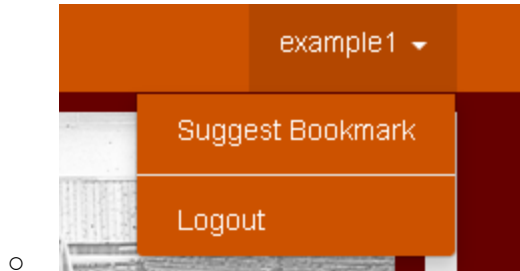
*Figure 22: Promoting a user*

- Testing to see if example1 is demoted:



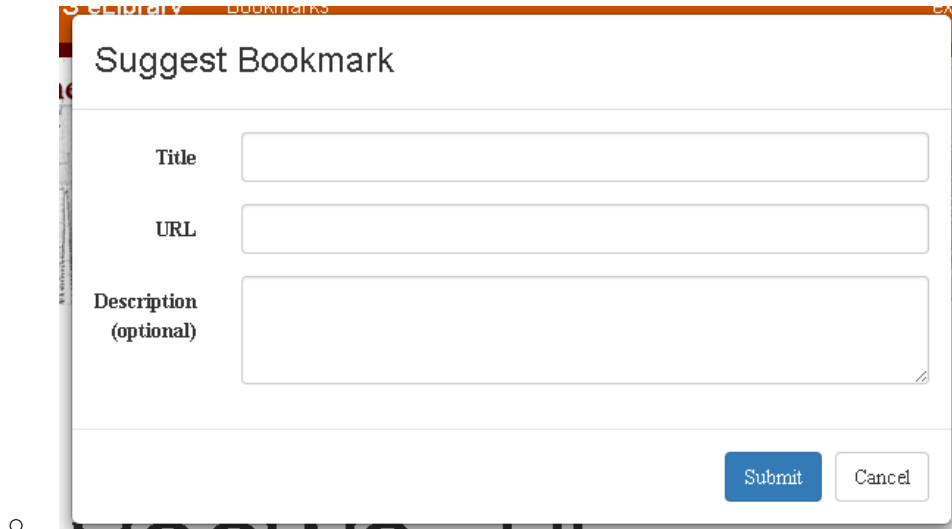
*Figure 23: Demoting a user*

- This is how you can verify the process of managing users through an admin user.
- Suggesting Bookmarks
  - Users can also suggest bookmarks which can appear in the bookmarks menu.



*Figure 24: User drop down*

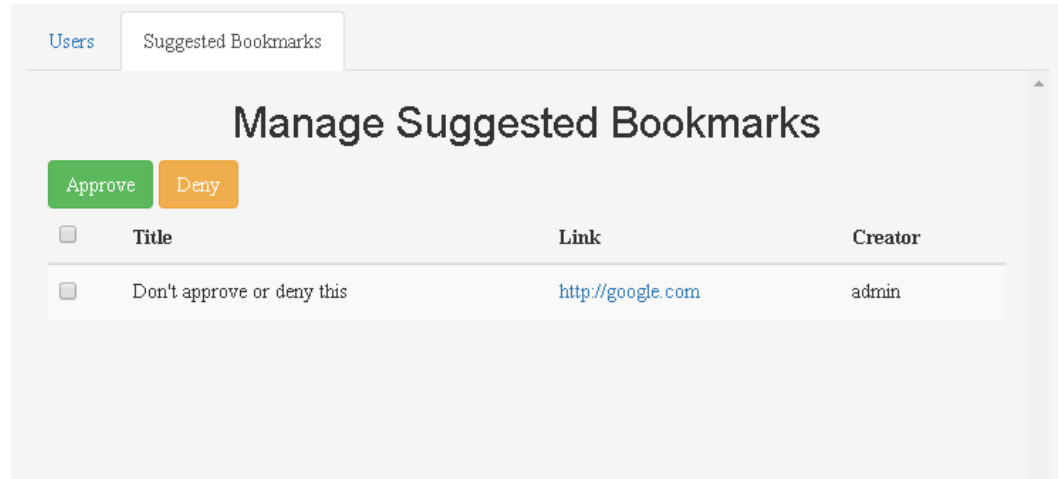
- First we can create a bookmark to suggest



*Figure 25: Suggesting bookmark window*

- Then the admin needs to either approve or deny the bookmark in the “Manage Suggest Bookmarks page”

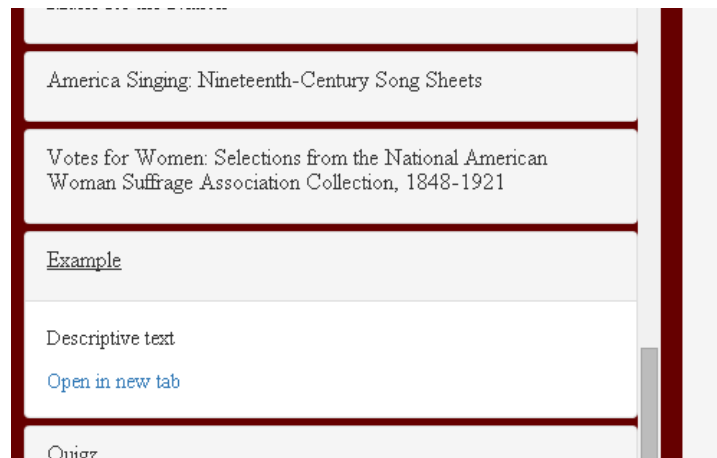




○

*Figure 26: Mange suggested bookmarks page*

- After the admin had approved or denied the page, it should appear on the bottom of the bookmarks menu.



○

*Figure 27: Suggested bookmark*

- This is how you can verify the process of suggesting a bookmark.

## **Future Work**

Currently, we do not track users and their visits to the site. Some potential future work here would be to collect some data and analyze it. If anyone wants to add code, in order to get it up and running you'll need an additional db.config file to specify where the database will be.

## **Acknowledgements**

### **Client:**

Paul Quigley  
Director, Va. Ctr. Civil War Studies

Department of History-0117  
Major Williams Hall, Rm 405, Virginia Tech  
220 Stanger Street  
Blacksburg, VA 24061  
United States

Email: [pquigley@vt.edu](mailto:pquigley@vt.edu)  
Work: 540-231-5331  
Local: 540-231-5331

# References

- [1] Using WAMP Server, Apache, PHP, MySQL on Windows. <http://www.wampserver.com/en/>
- [2] Bootstrap the most popular HTML, CSS, and JS framework. <http://getbootstrap.com/>
- [3] HTML enhanced for web apps. <https://angularjs.org/>
- [4] Agile software engineering management software. <https://trello.com/>
- [5] Code version control with GitLab. <https://about.gitlab.com/>
- [6] The iconic font and CSS toolkit. <http://fontawesome.io/>
- [7] Communicate through comments with Disqus. <https://disqus.com/>
- [8] The Virginia Center for Civil War Studies homepage. <http://www.civilwar.vt.edu/>
- [9] Agile software development with Scrum. <http://scrummethodology.com/>