12-8-2020

STEM Teacher Database

Veronica Buss
Western Michigan University, veronicabuss121@gmail.com

CS 4910 – Software and System Design II: Implementation and Testing Final Report

Veronica Buss, Desmond Cribbs, Emily Radocha

Table Of Contents

Table Of Contents 1

Abstract 2

Problem Statement 2
- Need 2
- Objective(s) 3
- Terms, Acronyms, Glossary 3
- Terms (Cont'd) 4

Problem Analysis and Research 4

Requirements 5

Standards and Constraints 6
- Applicable Standards 6
- Constraints 6

System Design 6

Testing 9

Results 11
- Realization of Requirements 11
- Realization of Standards and Constraints 12
- Testing Results 12

Future Work 12

Conclusion 13

References Appendices 15
- A. Project Management Plan 15
- B. Progress Reports 15
- C. Development Costs 22

Abstract

The College of Engineering and Applied Sciences (CEAS) Recruitment web application provides access to recruitment information for the Manager of Recruitment and Outreach and those who also use the spreadsheet file with their current data. This database is a functional database for the WMU college of engineering and applied sciences' recruiters to organize their data on STEM teachers from the feeder high schools of WMU. The app provides an interface for its users to filter and search the data they have compiled to create recruitment mailing reports. The main purpose of this app was to facilitate the retrieval and upkeep of their data. It was designed with our client's typical workflow in mind in order to make it as easy as possible for them to quickly find the data to be used in reports, as well as to use those reports in programs like Salesforce. Our work resulted in a fully functional database interface that gives the user the ability to create backups of the database, edit and add data, create reports, and manage other user access.

Problem Statement

Need

This project emerged from the need for a replacement for the current version of the CEAS recruiter's teacher database, which was a XLSX file. This file was the only way for student ambassadors to update the data and for recruiters to gather contact information for mailing purposes, but those who used and updated the master spreadsheet found that it was very difficult to keep the data recent and organized. Our client wanted to make it easier for others to be able to update it, so that when any recruitment information was sent out, it had the most accurate recipients.

Objective(s)

Our objective was to make an easy to use, easy to maintain, interactive database that was secure and efficient as well as include some extra functionality that would serve our client in his workflow. This includes the addition of the ability to create mailing reports based on the filters selected and ability to manage who has access to the database and to what extent. Since student ambassadors are using the app as well, we created separate views based on whether a user had an administrator or ambassador role set by the admin that gave them access. Ambassadors only have access to the database page where administrators have access to every part of the app. Our client also requested that we add the ability to backup the entire database to add extra protection for their data.

Terms, Acronyms, Glossary

- SQL
    - “Structured query language” is a language designed for managing data held in a relational database management system.
- Firestore
    - Cloud Firestore is a flexible, scalable database from Firebase and Google Cloud Platform.
- Interface
    - Anything that allows a user to control a software application. A good user interface provides a user friendly experience, allowing them to interact with the software or hardware in an intuitive way.
- Framework
    - A framework is a set of common and pre-built software building blocks that programmers can use, extend or customize for their specific purpose.
- Front-end
    - The front end of a website or application refers to the part that users see and interact with.
- Back-end
    - Back-end development refers to the behind-the-scenes of a website or application that isn't visible to a user.

Terms (Cont'd)

- Black-Box testing
    - Black box testing is a testing structure in which the tester is NOT familiar with the precise internal structure and implementation of the component being tested
- White Box Testing
    - White box testing is a testing structure in which the tester IS familiar with the internal structure and implementation of the component being tested.

Problem Analysis and Research

The few individual problems that we had to solve to complete the overall project were either ones that we had little to no experience with or ones that were out of our control, with which we had to get assistance with. The first problem we had to figure out was how to authenticate WMU logins. We contacted staff and worked with them to get the ability for the app users to log in with their personal WMU login credentials that we did not have access to. After we got assistance with login and hosting, we had to find a database that would suit the app's needs. It had to be secure and easily accessible so that it could be maintained after we graduate. We had prior experience with Google's Firstore, which is a no SQL database that allowed us to give our client's account full ownership once we were done. To further protect access to information such as the user information stored for admins to manage access to the app with, we had to find a way to create protected views where only certain authorized users could access. Our final problem that was something we had little to no experience with was creating reports from data through a browser. Upon research we found two libraries that helped us create and write data into XLSX files which is what the client required to be able to use those reports in their separate work software.

Requirements

Our group was able to meet every main requirement that our client had set for the scope of this project. We had established stretch-goals with our client from the beginning, and prioritized our main goals, as to assure that we would be able to provide a finished product that definitely worked for our client, as opposed to something that only somewhat worked for their needs.

Below are all of the requirements as they stood at the completion of the project:

- Establishment of two roles with different access to the database: administrator (admin) and ambassador
- Ability to easily access and update data.
- Filters populated by the data allow for users to filter by region, city, organization name, and department.
- Ability to create XLSX report files based on filtered data.
- A notes section for each teacher.
- A WMU alumnus status section for each teacher.
- Ability for administrators to create and restore backups of the entire database.
- Secure login where users are able to use their WMU credentials.
- Ability to search data by any field.
- Ability for administrators to be able to add and manage student ambassadors accounts

Standards and Constraints

Applicable Standards

The standards that were required for the web page to adhere to were WMU's conformance to the WCAG 2.0 Level AA guidelines as well as their own internal security standards, which were met with manual and automatic accessibility testing.

Constraints

The front-end constraints were that of each browser which did not have effect on the project as designed. The back-end constraints of the Firestore database are the restriction of reads that could be made from the database per day. There are a little over 8,500 rows that must be read to give the user full access to their database and so we had to make that large call as few times as possible during the app experience. We were able to stay under the daily limit for the free plan of Google's Firebase otherwise the next constraint would have been how much the client was willing to spend on database access and purchase more reads. There will be no cost of maintenance and there was no cost of production.

System Design

The system design for the CEAS Interface stemmed around the core requirements and features our client had set for us. This included making separate views for the different roles who have varying levels of access to the database, namely: a view for the administrators of the database, and a view for the student ambassadors. Both of these roles were to log in with their WMU credentials. In addition, the distinct sets of features available to interact with the database were taken into account, and different pages created as a result.

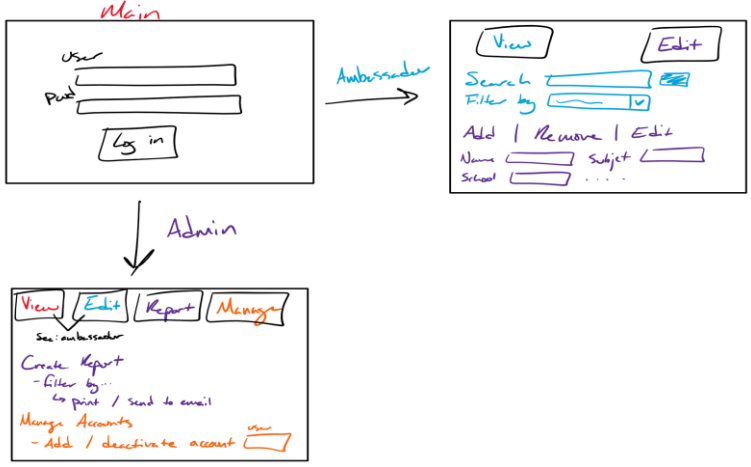
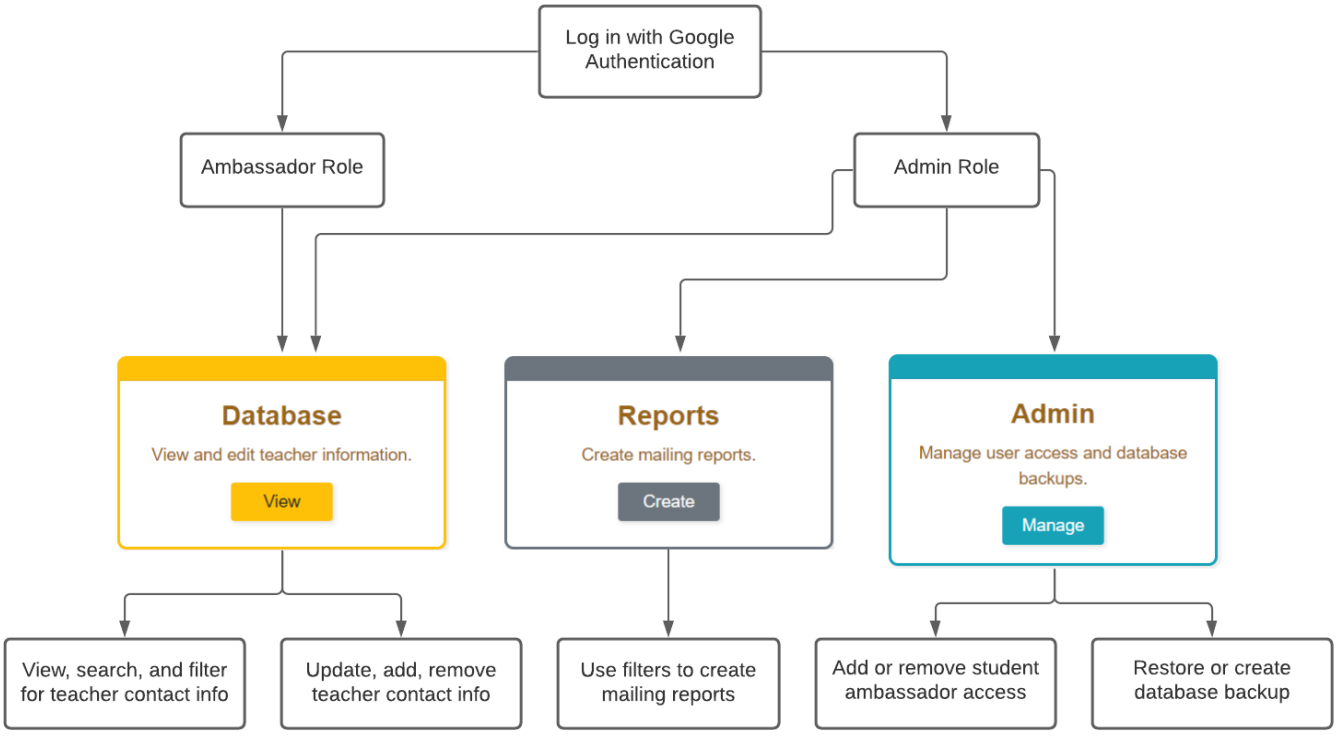


Figure 1. Initial role differentiation and page layout

Figure 1 shows the first iteration of the system design, where the login and distinction of roles had been established, however the specifics on the differentiation between pages had not yet been decided. While some of the buttons were in place representing future implementation of the intended features, this was a very preliminary draft which was used as a general idea confirmation with our client.

Figure 2. Final System Design Diagram

The final system design came after distinct pages had been decided and groups of



features had been put together. As is shown in Figure 2, first users use their WMU credentials to

login with google authentication. If the user is an ambassador, they are only able to see the Database page, and if the user is an administrator, they are able to see the Database, Reports, and Admin pages. From the Database page, users can view, search, and filter for teacher contact information, in addition to being able to update, add, or remove information. From the Reports page, users can use the filters to create mailing reports, and from the Admin page users can create a database backup, or restore the database from the backup.

In terms of the technical system design, the javascript frontend framework Vue was used, with main pages being the Home page, Database page, Admin page, and Reports page. Modal components were used for all of the delete confirmations, add information popups, the edit teacher prompt, and prompts related to the database backup.

In terms of the data itself, the CEAS Recruitment web application interacts with the data stored on google's firebase in two distinct tables: teachers and users. The Database and Report page interact with the teachers table, allowing for usage of the data in the filters as well as adding, updating and removing data directly from the database. The admin page interacts with the users table in adding and removing access for student ambassadors or other admins.

In addition, for authentication in order to access the database at all, an admin needs to add the user to the database on the Admin page. From there, an email is sent to the user inviting them to join the database. Only once the user clicks the link to confirm their email and they are all set to go can they then access the database. An example of the automated sign in email sent to users is shown in Figure 3.

Figure 3. Invitation email to CEAS Recruitment database

Testing

Testing is a necessary and crucial part of any software project in order to ensure a smooth workflow and work out any kinks that may slip by during the developmental process. The types of testing performed in the testing plan included:

- Black box testing - Of the members of the group, one member mainly worked on the frontend, one member mainly worked on the backend, and one worked on the authentication / login. Since various members of the group worked on different parts of the project, this testing was performed by members who were working on opposite parts of the project, namely login and frontend/backend.
- White box testing - In order to have a working web application, the frontend and backend pieces need to be in coordination. However, while the members who worked on these parts of the project had a better understanding of what was going on on the other side than the login member, this type of testing was valuable in catching various bugs.

- Accessibility testing - Automatic testing to the WCAG 2.0 Level AA guidelines were performed using the WAVE (Web Accessibility Evaluation Tool) from WebAim and Utah State University which resulted in needing to add aria-labels to various elements and ensure the color scheme had enough contrast for readability. An example of the WAVE testing output is shown in Figure 4. In addition to automatic testing, manual testing was also performed. The Vue frontend framework made it really easy to comply with this standard, as focus tabbing and other features are built-in to their components. As an added bonus, a developer's family member who has a disability gave her feedback on one page of the website, further affirming that the page is indeed accessible.

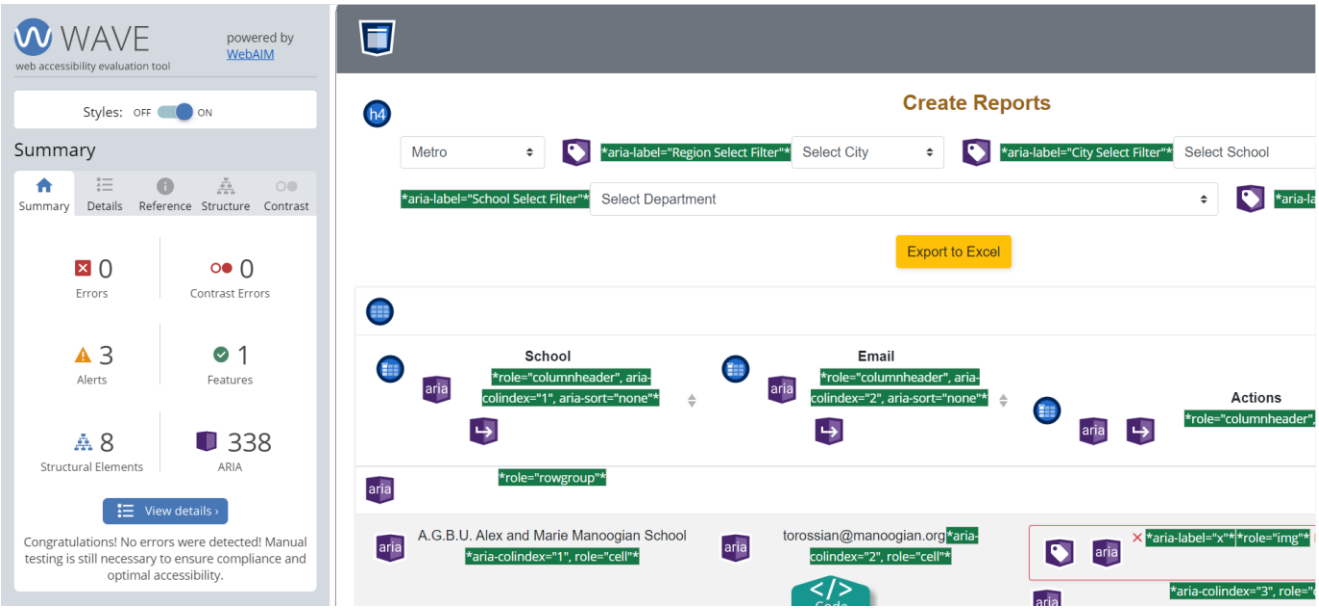


Figure 4. The WAVE Accessibility Evaluation Tool in action

- Usability testing - This testing was performed by family members and friends who were unfamiliar with the project to study how usable the web application was to an outside pair of eyes. This testing resulted in some valuable feedback in terms of the placement of buttons, descriptions of elements, and color schemes.

In addition to the coordinated testing, the group had weekly progress meetings where we would discuss and log any bugs we noticed, as well as biweekly mentor meetings and monthly client meetings, where we demonstrated the current state of development of the web application. This constant feedback from especially the client helped to ensure the group was on the right path to satisfy requirements and have a working product.

Results

Realization of Requirements

As mentioned previously, the group was able to realize all of the main requirements including establishing two roles with different access to the database: administrator (admin) and ambassador, the ability to easily access and update data, usage of filters populated by the data that allow for users to filter by region, city, organization name, and department, the ability to create XLSX report files based on filtered data, a notes and WMU alumnus status section for each teacher, the ability for administrators to create and restore backups of the entire database, a secure login where users are able to use their WMU credentials, the ability to search data by any field, and the ability for administrators to be able to add and manage student ambassadors accounts.

While all the main goals were met, the group was not able to achieve the stretch goal of the project, which was creating a web form to send out to teachers to input their own data. This feature would have allowed for teachers to come to a separate webpage on WMU CEAS' website and submit their email which would send them a link to a webform where they could securely enter in their own information to opt in to recruitment information. This would have then been automatically inserted into the database as opposed to student ambassadors inserting

the data manually, however since we would have had to make sure that the input was scrubbed and secure for this to be a secure, working feature, we left it off.

Realization of Standards and Constraints

We were able to meet WMU's conformance to the WCAG 2.0 Level AA guidelines with manual and automatic accessibility testing. With the limitations set by google firebase in terms of how many times we can read from the database per day, we developed a design that would stay under that limit. This was done by requesting a database read only when completely necessary so that we won't exceed the daily limit.

Testing Results

The black and white box testing performed throughout the process helped us to uncover some bugs and strengthen any weak points that otherwise might not have been discovered. For example, white box testing revealed some inconsistencies in the modal popups on the admin page for adding an account, as well as for the edit account inline feature, which was why we decided to switch to an edit account modal popup instead. The accessibility testing led us to be confident that we had obtained conformance to WMU's accessibility standard, and the usability testing we did ensured that anyone in the future who is new to the database can intuitively navigate through the pages and perform necessary actions.

Future Work

Our application was developed using the free "Spark" plan of firebase and had limitations in relation to document usage, stored data, and available cloud functions. Using the paid "Blaze" plan of firebase, cloud functions could be integrated into the application which will open up

more user management and data manipulation options based on events triggered by firebase features. The maximum amount of stored data will also increase along with the number of document reads, writes, and deletes. Therefore, the Blaze plan would increase the overall work power of our application. As stated before, the stretch goal feature to allow for teachers to enter in their own information for recruitment information has not been added. If implemented, ambassadors would not have to manually type in the teacher's information and it would instead be stored automatically.

Conclusion

The CEAS Recruitment web application offers a simple and accessible web form that produces an organized view of recruitment information for the user. The idea behind this came from the need for a more functional database of the current XLSX file that was being used for the recruiters' teacher information. This file was the only thing used when a student ambassador needed to update the data or when a recruiter needed contact information for mailing purposes. Consequently, ambassadors and recruiters would experience difficulties in terms of editing and organizing the teacher information. Therefore, our client wanted to make these tasks easy to complete in order to ensure that when any recruitment information was sent out, it had the most accurate recepients. Additionally, the ability to create mailing reports based on filters, to manage user's authorization, and to create a backup of the entire database.

We were able to satisfy all of the main requirements our client has set for the project. This included integrating two roles that determined how the user accessed that database, updating data stored in the database, filtering data based on related categories, creating XLSX report files based on filtered data, creating and restoring database backups, secure login with

WMU credentials, and administrative user management. We also established stretch goals with our client, but prioritized the core features first to preserve their quality and effectiveness.

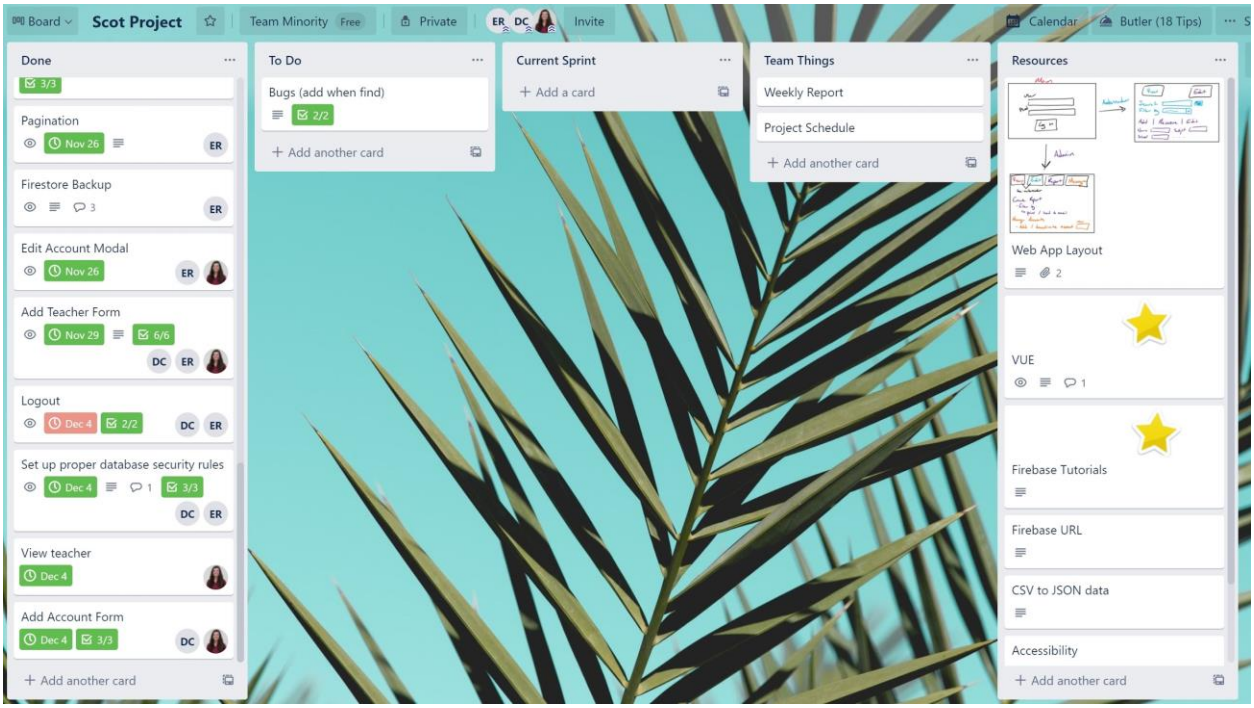
Multiple forms of testing practices were used to ensure the project has a smooth workflow and to eliminate any error or bugs that may have been initially overlooked. Black box and white box testing proved most useful in detecting any bugs. Using the web accessibility evaluation tool, we were able to perform automatic and manual accessibility testing so that WCAG 2.0 Level AA guidelines were met correctly. For usability testing we had received valuable feedback from friends and family members who had no prior information on the overall design of the project, resulting in changes to the color scheme and button placement.

During the final developmental stages of the project, we integrated all the required features and made the necessary edits to maintain the overall workflow. We were unable to add the stretch goal feature for automatic teacher additions. This would allow a teacher to securely add their information to the application only after receiving an email link from the WMU's CEAS website. Furthermore, the CEAS Recruitment web application creates an easily accessible and functional database which provides a more efficient workflow for the Manager of Recruitment and Outreach.

References Appendices

A. Project Management Plan

The team utilized Trello.com to manage their task and each week's sprints to make sure that task assignments were clear and to keep everyone on task and focused on the end goal.



B. Progress Reports

CS 4910 – Software System Development and Design II: Implementation and Testing

Project Progress Report 1

Project Information:

Team Members: Veronica Buss, Desmond Cribbs, Emily Radocha
Client: Scot Conant
Advisor: Dr. Hong
Report Date: 10.8.20

Team Activity Report:

Since our last meeting (10/1) we have been setting up the tools for our project. Emily set up the VUE framework and integrated the Firestore database into it. Desmond set up Firebase authentication to begin working on the login page. Veronica has set up Selenium to work with our project so we may run tests as well as began integrating VUE Bootstrap.

Client Interaction Report:

We have not met as a team with Scot since last semester, we are now communicating via email and zoom if more communication is necessary. Emily met with Scot to get the current excel spreadsheet where they have kept all of their data that they want transferred into the database. Through emails we have evaluated how Scot wishes to filter and use the database as well as discussed which fields he would like to change or add.

Milestone Review:

This phase of the project is the setup phase. We have decided on all the tools we are going to use and have begun implementing them. All frameworks and other tools will be fully set up and ready to use with our project by the end of this week. We are also in the process of finishing the implementation phase. Connecting our framework to our database, authentication service, setting up bootstrap, and our testing program.

Issues (or stories):

- As an admin / ambassador I can log in to the database with my WMU credentials
- I can set filters on the database to see the data I want to
- The web page is set up neatly, is intuitive, and is easy to navigate

Problems and Risks:

We are having trouble figuring out a data model for efficient storage and retrieval of all the data. To address this, we will be talking to our mentors who have experience with this topic.

CS 4910 – Software System Development and Design II: Implementation and Testing
Project Progress Report 2

Project Information:

Team Members: Veronica Buss, Desmond Cribbs, Emily Radocha
Client: Scot Conant
Advisor: Dr. Hong
Report Date: 10/15/20

Team Activity Report:

After last week's setup, we have developed additional back-end support to our project. Emily worked on the database page backend and added data. Veronica made the admin home page, manage accounts page, and login skeleton pages and worked with Emily on the database filter. Desmond added functionality to the login in page and integrated authentication requirements for each page.

Client Interaction Report:

We have not met since the last report, we will be emailing him tonight to upgrade the project's plan on Firestore from free to the Blaze plan so we can perform more reads / writes during development as well as be able to add a database backup functionality to the app. We will be providing progress to the client once we have all the basic functionality set up.

Milestone Review:

We have setup all the tools and frameworks that we plan to use. We are now working on implementing functionality, starting with being able to work with the data in the database and login and then we will be moving on to creating reports and managing user data. We are on schedule and doing well.

Issues (or stories):

I am able to add and remove data from the database.
I am able to edit data and see the changes in real time.
I am able to log in and dependent on my role, I am shown different views.

Problems and Risks:

We are not sure how to go about coordinating student ambassador's logins with their WMU credentials. To mitigate this issue, we will be getting in contact with WMU Faculty Specialists and asking for their advice.

CS 4910 – Software System Development and Design II: Implementation and Testing
Project Progress Report 3

Project Information:
Team Members: Veronica Buss, Desmond Cribbs, Emily Radocha
Client: Scot Conant
Advisor: Dr. Hong
Report Date: 10/22/20

Team Activity Report:
Desmond has gotten firebase authentication working so we can log in to the site with our data. Emily has added functionality to the database as well as the manage accounts page so data can be added, deleted, and edited. Veronica has changed and made additions to the database table page as well as the manage accounts page.

Client Interaction Report:
We have a video meeting set up with our client for tomorrow. We have been emailing him with questions and will be giving him a look at what we've gotten done tomorrow as well.

Milestone Review:
We are implementing functionalities piece by piece. After we got the structure of our database table, we added functionality so the user can interact with it, and made changes to the frontend to fit with the backend. After we feel full functionality has been reached with the database, we will be moving on to features such as searching, creating reports, and etc. We are on schedule and are gaining momentum as we continue to work on this together.

Issues (or stories):
Our only issue that we cannot do by ourselves is login authentication with WMU's credentials and integrating our web app onto WMU's website. We are in contact with the people who can help us do that, and are awaiting their help.

Problems and Risks:
No problems have arisen, we are on schedule and on course.

CS 4910 – Software System Development and Design II: Implementation and Testing
Project Progress Report 4

Project Information:

Team Members: Veronica Buss, Desmond Cribbs, Emily Radocha
Client: Scot Conant
Advisor: Dr. Hong
Report Date: 10.29.20

Team Activity Report:

We've continued fleshing out pages as we continue adding functionality and then making the UI match throughout the pages. Desmond is working on being able to add and delete authenticated users. Veronica is working on the last page we need to create, the create reports page. Emily is finishing up search and filter functionality. We discussed who else we need to contact to get help with getting the security and hosting set up. We are currently talking to Chris Rand and Richard Burton to figure out what needs to be done to get our website hosted, linked to WMU's authentication, and align with security protocols.

Client Interaction Report:

We met with Scot last Friday and showed him how the website looks so far and got his feedback on what we could change / add for him. We also discussed his workflow so we can optimize the creation of reports so that it works as he wants it to.

Milestone Review:

We are on schedule with completing the tasks we set for ourselves each week. We are in the middle of creating the base functionalities for the entire website, then once that is done, we will move onto fine tuning that functionality after testing it for the size of data that will be used.

Issues (or stories):

- I am able to search the database for data I want to see
- I am able to filter the data to narrow down what data I am seeing
- I am able to add data to the database
- I am able to add a user to have access to the database
- I am able to edit or remove teacher or user data

Problems and Risks:

We are having issues finding a way to add and delete authorized users. We have a backup plan in case this problem takes too much time to solve or ends up being unsolvable. We also are waiting on the people we've contacted, mentioned above, and have asked to set up a meeting to get some clarification and if we receive no response we will contact Jason and our client.

CS 4910 – Software System Development and Design II: Implementation and Testing
Project Progress Report 5

Project Information:

Team Members: Veronica Buss, Desmond Cribbs, Emily Radocha
Client: Scot Conant
Advisor: Dr. Hong
Report Date: 11/5/20

Team Activity Report:

We've continued to make progress on adding functionality to our key features. Emily added the ability to create an excel spreadsheet based on filtered data. Veronica worked on the view/edit page add/edit teacher modals and the styling for the filters. Desmond created the firebase emulator for testing cloud functions.

Client Interaction Report:

We are meeting with our client next week to show him our progress and ask a couple questions to tailor the experience for him. We will be demonstrating the functionalities we have added for him and get feedback.

Milestone Review:

We are nearing the end of adding all the functionalities required by the client and that make the app function as a whole. This part of the project should be fully completed within the next two weeks. After this part we will begin any stretch goals and finalizing details.

Issues (or stories):

- I am able to create an excel report based on filters I set.
- I am able to add a teacher's information into the database.
- I am able to log in using Google's authentication provider.

Problems and Risks:

There haven't been any problems that have set us back. We are still collaborating with WMU staff to get log in abilities and host our pages on the CEAS website. This is the largest risk in terms of costing us time if we can't get it, although it is not a large enough risk to stop the project as we have people to help us if we run into any problems.

CS 4910 – Software System Development and Design II: Implementation and Testing
Project Progress Report 6

Project Information:

Team Members: Veronica Buss, Desmond Cribbs, Emily Radocha
Client: Scot Conant
Advisor: Dr. Hong
Report Date: 11.12.20

Team Activity Report:

Emily made adjustments to and bug checked the filters to make sure they worked with more data. Desmond finished integrating Google sign in

Client Interaction Report:

We met with Scot this past Monday to show him a live demo of what we have so far. We discussed how he would like the mailing reports to be formatted and what our stretch goals are and how we're progressing.

Milestone Review:

We are finishing up functionality and moving on to security requirements, we should be finished entirely with functionality in two weeks. We are also fairly there in terms of security requirements, once we check in with Richard Burton we will know whether we need to manually hash our data or not given Firestore's security.

Issues (or stories):

- I am able to login and view page content based on my role
- I am able to filter all data within the database dynamically
- I am able to add a teacher with all their information into the database
- I am able to edit a teacher's information

Problems and Risks:

We do not have any problems at the moment. Our three risk factors currently are the things we do not have full control of which are:

1. Security
2. Hosting
3. Login

Like we said security is almost done, we just need verification from Richard. We also need to know if there's anything we need to do specifically to help get the app hosted on wmich.ceas. We also need information from Richard to know if we need more than Google's auth provider which we currently have set up to log in with.

CS 4910 – Software System Development and Design II: Implementation and Testing
Project Progress Report 7

Project Information:

Team Members: Veronica Buss, Desmond Cribbs, Emily Radocha
Client: Scot Conant
Advisor: Dr. Hong
Report Date: 11.19.20

Team Activity Report:

Veronica added the edit teacher modal and Emily added the functionality to the modal. Desmond has been working on the add teacher form validation and is continuing to work with Richard to implement the log in functional. Emily is also continuing to test and fix any bugs that arise.

Client Interaction Report:

We have not met with Scot as he was on leave for a week.

Milestone Review:

We are still finishing up functionality that will be finished within the next week. We are on schedule with functionality and behind schedule with log in because we are waiting on a response from Richard Burton.

Issues (or stories):

- I can edit a teacher's information to update the database
- I can add a new teacher to the database
- As an admin I can edit an account's information / access

Problems and Risks:

So far, no problems have arisen, we have confirmation from Brandon who is helping us host our site that he has a backup plan for us in case they can't get it hosted with CEAS by Dec 8th.

C. Development Costs

There were no monetary costs to the client or to the team to complete the project.