

Design Document for: Scalable Web Application Framework for Monitoring Energy Usage on Campus

Daniel Schroeder, Aubrey Thenell, Parker Bruni



Abstract

The purpose of this progress report document is to outline the progress made on the Scalable Web Application Framework for Monitoring Energy Usage on Campus project over the past ten weeks. Provided in this outline are chronological digests of the accomplishments and problems presented each week, our project's goals and purpose, the current status of our project, and a retrospective of the term.

CONTENTS

1	Purpose and Goals:	3
1.1	Purpose	3
1.2	Goals	3
2	Past Weekly Events:	3
2.1	Week 3	3
2.1.1	Activities	3
2.1.2	Problems and Solutions	3
2.2	Week 4	3
2.2.1	Activities	3
2.2.2	Problems and Solutions	4
2.3	Week 5	4
2.3.1	Activities	4
2.3.2	Problems and Solutions	4
2.4	Week 6	4
2.4.1	Activities	4
2.4.2	Problems and Solutions	4
2.5	Week 7	4
2.5.1	Activities	4
2.5.2	Problems and Solutions	5
2.6	Week 8	5
2.6.1	Activities	5
2.6.2	Problems and Solutions	5
2.7	Week 9	5
2.7.1	Activities	5
2.7.2	Problems and Solutions	6
2.8	Week 10	6

		2
2.8.1	Activities	6
2.8.2	Problems and Solutions	6
3	Retrospective:	7
4	Current State of the Project:	8

1 PURPOSE AND GOALS:

1.1 Purpose

Our project is to create a web application to monitor energy use on Oregon State University's campus. The application should serve all the requirements outlined by the client and be easy to use for users of all experience levels. Some specific functionalities that our application should contain are:

- Receive data from Obvius AcquiSuite data acquisition servers and process this data into interpretable graphs.
- Allow administrative users to add buildings and meters to the database as monitoring efforts expand to more buildings on campus.
- Allow users to create unique dashboards and dashboard collections in an effort to organize data into related subsets.
- Have a public facing interface where administrators can produce content for anyone to see.
- Contain modular components with individualized functionality and the ability to share data across components.
- Update graphs being displayed as new data is received.
- Undergo usability testing and produce an interface that is user friendly and easily navigable.
- Embrace AngularJS concepts to inject content to the page as new requests are made.

1.2 Goals

The goals of this term were to research and document the different technologies and methods needed to implement the required functionality of our application. This included researching specific technologies and frameworks as well as designing the architecture of our application's subsystems data models. Our designs revolve around highly modular components that split up the applications overall functionality into smaller components and services. We would like to begin implementing our designs and applications over winter break and start winter term with a good code base already constructed.

2 PAST WEEKLY EVENTS:

2.1 Week 3

2.1.1 Activities

This week involved a general setup of resources for the rest of the term including a Github repository, OneNote Notebooks, and a Slack channel for group communication. We also met with our client, Jack Woods, at his office in the Office of Sustainability on campus to discuss details about the project and their visions for the web application.

2.1.2 Problems and Solutions

2.2 Week 4

2.2.1 Activities

In week four, our team finished the Problem Statement final draft and submitted on Github and OneNote. We also emailed our client requesting a list of requirements he and his team want to see in the final application to

begin working on the Requirements Document assignment. Daniel met with Professor McGrath to look over the requirements list, which Kevin approved with no need for consultation or negotiation. This was the first week we were required to meet with our TA, Ben, where we discussed the structure and organization of directories in our group repository and how to create a logical labelling system. We also made a plan of action to begin researching the MEAN stack and its different frameworks.

2.2.2 Problems and Solutions

2.3 Week 5

2.3.1 Activities

This week, our team got a rough draft of the Requirements Document completed and submitted on Github and OneNote. We also met with our client to discuss further details about the components listed in his requirements list.

2.3.2 Problems and Solutions

We had to create a Gantt chart in excel and insert an image into our Requirements Document because we could not get the Gantt Latex package to work correctly.

2.4 Week 6

2.4.1 Activities

This week we got a Latex Gantt package to work in our Requirements Document and submitted final draft to Github and OneNote. We also contacted our client to review and approve our Requirements Document and send a confirmation email to Kirsten Winters and Kevin McGrath. We were very content with the communication and rate of response from our client; it simplified our workflow and made accomplishing our tasks very trivial. Later in the week, our group assigned three components to each member to research for our individual Technology Review documents. Our TA meeting revolved around clearing up the final questions about the Requirements Document and discussing our current actions required model. Included action were to split up components for the Technology Review, get a Gantt package to work in Latex, and finish final draft of the Requirements Document which were all satisfied.

2.4.2 Problems and Solutions

2.5 Week 7

2.5.1 Activities

This week, Daniel made a simple web socket MEAN stack application to receive a DATE object from the application server every second and render the new data to the page (effectively creating a clock). Daniel wanted to start small scale MEAN stack applications to begin implementing some of the strategies being reviewed in the Technology Document which will eventually be implemented in our final application. Daniel also began looking at dynamic D3 graphs which will be used as the visualization framework in our final application and added dynamic D3 bar graph template to the web socket application. Our group also split up the components for our Technology Review documents. The apportioning of technologies for the tech review were: Daniel:

- Visualization frameworks

- Means of Authentication
- Front-end Frameworks

Parker:

- Database Framework
- Database Host
- Back-End Framework

Aubrey:

- Structural Frameworks
- Server-side Web Application Framework
- Web Hosting

2.5.2 Problems and Solutions

We have had trouble getting our client and his office to set up an AWS account to host our application. He ensured us that he would get it taken care of, but we are currently stuck with only local development strategies in the meantime.

2.6 Week 8

2.6.1 Activities

This week our group finished the individual Technology Review rough drafts and began working on the final drafts for next week. We underwent extensive research on specific technologies that would be incorporated in our application and generated good documentation of examples and resources to help drive future development.

2.6.2 Problems and Solutions

In the tech reviews, we were originally going to write about password hashing algorithms with the assumption that we were going to create our own authentication system. After discussing this issue with Kevin McGrath and our TA, Ben, we were guided towards outsourcing our authentication and researching different ways of implementing an authentication system rather than hashing algorithms.

2.7 Week 9

2.7.1 Activities

This week was short-lived due to the Thanksgiving holiday, but activities included submitting the final drafts of the Technology Reviews and beginning work on the Design Document. After the holiday, we created a template Latex file for the design document and began reading the IEEEtran standard for the SDD. We created a general outline for the sections we needed to cover in our design document based off the content from the IEEEtran description.

2.7.2 Problems and Solutions

The IEEEtran document was extremely verbose and complicated to understand upon initial review. After discussing this issue with Kirsten Winters, she advised interpreting the IEEEtran document as an expansive example of everything that could be included in a design document and to focus on parts that best correlate to our project. Composing the design document also posed significant concerns regarding how we are going to design all the subsystems of our application efficiently.

2.8 Week 10

2.8.1 Activities

This week was extremely helpful in understanding all the different components that our application will have and designing modular components and services that will provide the necessary functionality for our application. While designing the application's architecture, our group considered the different subsystems, data models, and elements that are going to be needed in the final product. After designing the architecture, we met with our client to discuss our ideas and gather feedback. This discussion proved to be extremely helpful and we included his recommendations in our final draft of the Design Document. Daniel created a high-scale MEAN stack application to begin implementing a lot of the features from the Design Document and Technology Reviews like:

- Using Google's authentication API with Passport.js to store users into a database and create user sessions with passport's authentication services
- Serving content to the view container by injecting routes to a parent scope variable in the master page by triggering an AngularJS injection to render a new view
- Creating a template for our block objects using Bootstrap 4 cards
- Creating modular controllers and services for different components and functionality
- Dynamically changing the contents of components when new data is added to the database

2.8.2 Problems and Solutions

We had some troubles with explicit definitions of terms in our requirements from our client. We had multiple conversations and messages from our client where he used the terms "dashboard" and "page" interchangeably. After meeting with him, we cleared up the confusion and were able to establish a finalized dictionary of terms. Another issue we faced was the design of data models that presented themselves in the Design Document and determining how we were going to efficiently store and manage our different data and entities.

3 RETROSPECTIVE:

Positives	Deltas	Actions
Our team is very responsive to group communication via our Slack channel	Need to establish more in-person meetings to discuss abstract ideas in greater detail	As winter term begins, we will be meeting more in person during the development of our application
Client was very easy to communicate with and was quick to accomplish tasks we needed	We would like to meet at least once a week over the development process to share progress and acquire feedback	Get client on board with meeting next term and find a time that satisfies every member's schedule
Attained a high level understanding of our application throughout the course of this term	We could do better at discussing abstract ideas about our application's architecture so all members are in agreement	Meeting in person more to discuss, in detail, the abstractions of our application
Got small-scale applications working locally to begin working with concepts and frameworks that will be used in our final application	Could have more group members up to speed on MEAN stack development	Meet in person to discuss techniques that team members produce individually to ensure understanding across all members

Table 1: A retrospective of the past ten weeks.

4 CURRENT STATE OF THE PROJECT:

We have began developing local, small-scaled test applications to begin implementing some of the frameworks we will encounter in the final product. These include MEAN stack applications that:

- Authenticate users with Googles oAuth 2.0 API and keep track of user sessions
- Use websockets to send intermittent time data from the server to the page
- Statically render different views to the content container with Angular 1 ng-include and passing routes from navigation items
- Share data across different modular components
- Dynamically change content as users submit forms and insert data into the databases

These small scale implementations have guided the construction of our different documents this term, like the technology review and design document, by providing hands-on experience of the actual construction of different system components. These test applications required more explicit research than what was required for the written documents in order to produce a functional application rather than an abstraction of a working product. We will begin to compile and transpose these smaller solutions into the main source code of the application once we begin development over winter break.