CS Capstone Problem Statement

October 9, 2017

A Scalable Web Application Framework for Monitoring Energy Usage on Campus

Prepared for

Oregon State Office of Sustainability

Jack Woods
_____   _____
Signature   Date

Prepared by

Group 57
The Dream Team

Daniel Schroeder   _____   _____
                          Signature   Date

Aubrey Thenell   _____   _____
                          Signature   Date

Parker Bruni   _____   _____
                          Signature   Date

Abstract

The purpose of this project is to create a better implementation to the current web application used by the sustainability office to monitor energy use around Oregon State University. This web application will be a scalable MEAN stack app that will gather data from various AcquiSuites sensors around campus and create informative, useful visualizations. This application will help to visualize energy usage around campus and incentivize certain buildings or departments to acknowledge their energy use in order to reduce the overall usage around campus and meet the Oregon State University Carbon Neutral goal of 2025.

CONTENTS

1 PROBLEM DEFINITION

Oregon State University is revered for its achievements in sustainability and constantly strives to reduce its carbon footprint through advanced technology, research, and innovation. The Office of Sustainability collects energy usage data from AcquiSuite sensors in numerous buildings around campus and stores this data in a database. They use this information to populate a dashboard they use to visualize charts and graphs of buildings energy use. The Office of Sustainability outsourced this project to a company called Lucid who are responsible for the creation and implementation of this energy dashboard. The issue is that Lucid's service is sufficiently expensive that scaling the application and growing the dashboard is extremely costly.

In 2008, Oregon State University developed a plan to reach carbon neutrality by 2025 through renovations, new technology, and better practices. Working towards their goal, Oregon State has financed projects such as the Solar Farm, Kelley Engineering Center, and the Johnson building that have implemented modern green technology and infrastructure to reduce, reuse, and renovate the way Oregon State uses energy. The Office of Sustainability believes that part of the battle is awareness and exposure. In order to help reduce energy use and emissions on campus they have financed this dashboard that can help certain departments and specific buildings see the cold, hard numbers detailing their usage. They believe that this sort of awareness will get people to be more mindful of their energy habits and work towards reducing their personal carbon footprint which, in turn, will benefit the entire campus.

The Office of Sustainability chose to reach out to the senior capstone program in the college of computer science and request a student designed and implemented dashboard that could perform the same tasks as Lucid's software, without the expenses. By allowing the computer science department to take over this project, the Office of Sustainability will be able to have an internally sourced web application that can be modified to whichever specifications they desire. It will also allow the hundreds of thousands of dollars that was previously spent on Lucid's services to be spent on more sustainable projects for the entire University and help Oregon State reach its carbon neutrality goal of 2025.

2 PROPOSED SOLUTION

This project will include building a full stack web application capable of utilizing the AcquiSuite API to retrieve and store energy data. With access to the building data, we will have to build up a smooth UI that can dynamically (and hopefully RESTfully) create graphs and visualizations to display building energy use.

Our group proposes to design, build, and implemet a scalable MEAN stack web application that incorporates multiple visualization tools and data manipulation capability. Overall, we would like to present a clean, simple interface where the user can apply filters and see different data sets. We also want the ability to scale the application through adding more buildings and AcquiSuite data to the database as more buildings become monitored.

We plan to utilize a lightweight graphics framework like d3.js or vis.js to create charts and graphs within our web application. Additionally, we would like to have the option of displaying multiple types of graphs and charts for multiple data sets to give the user a well-rounded visualization experience. For example, a line chart that shows energy use over time, and a pie chart that compares buildings or timeframes. Having a multitude of different data visualization techniques is essentially in creating an effective visualization tool that can be interpreted for various purposes.

Our group also plans on implementing Angular.js, to create dynamic views within the pages of our web application. We can utilize the Angular framework to allow data binding and static visualization rendering. This will allow us to display multiple charts and data sets without having multiple page reloads and waste server time or browser load time.

Angular will also allow us to cache data sets and user interaction to speed up any and all client/server interaction.

This web application will follow strict MVC guidelines to ensure that the Universities data and security is help to the utmost importance. The Express framework will act as our controller that will link the database with the web application's view and maintain permissions for security and data integrity.

3 PERFORMANCE METRICS

The performance metrics for this assignment will include the ability to scale the web application with the addition of more building and AcquiSuite sensors. This is essential to providing a sufficient product to the Office of Sustainability. Ideally, we would like to have an admin center where a user can add buildings to the database along with the AcquiSuite id number and the web application will handle creating a database entry for use and adding that building to the dashboard. This web application should also have performance metrics based on load time and server load. If loading a graph takes 10 seconds, there is probably an issue within the code that could be fixed to generate graphs faster. We would also like to have the visualization generation be highly dynamic and utilize Angular's framework to data bind to the page rather than belabor the back end with server calls and requests.

There are project requirements that have yet to be set, but if time prevails, we would like to provide two separate portals within our web application. One portal for a general user to view and filter data. Then a second portal with admin rights where employees of the Office of Sustainability can add new buildings or manipulate what sort of graphs are available in the dashboard. This would require a lot of overhead with permission checking and alternate page views, but would be ideal for a professional web application.

Even though it is subjective, another possibility for performance metrics could include UI and design. After all, we are creating a product for a particular customer, so designing to their likes and desires is crucial.

As a minimum viable product, I would like to have a web application with all the currently monitored building stored in the database and access to the AcquiSuite sensor's information. In addition, I would like to have static pages that can generate different visualizations based on the provided filters and have them data bound through the Angular framework onto the specific region designated by our page design.