Project Progress Report
CS 410 – Text Information Systems
N. Wood (nadiaw2)

Search engine for indoor environment data using ElasticSearch

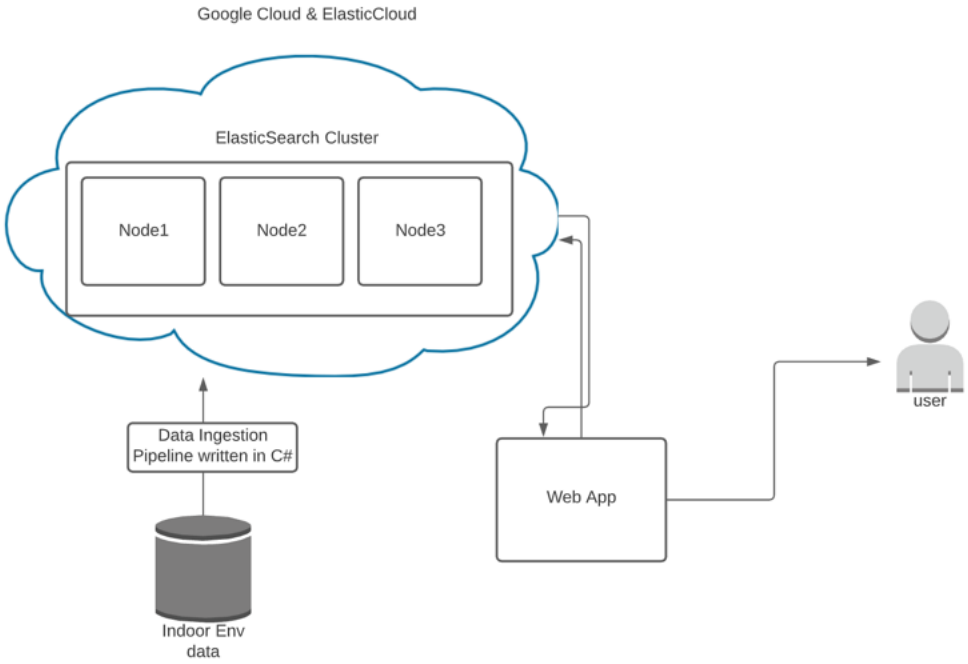
NetID: nadiaw2. I will be working on this project individually.

Abstract:

In the age of sensors, devices and platforms collecting millions of datapoints every second, it comes necessary to be able to sift through all the data to develop insights efficiently Big data offers the solution for analyzing large amount of data and using the technique of Elasticsearch, access to data can be made faster.[1]

I will be creating a web application to use ElasticSearch to search content from a set of documents of environmental datapoints collected by sensors (indoor air, humidity, temp etc). Currently, it is difficult to search for data in a RDMS database and it takes significant time using traditional SQL queries. The project will take the data transfer it to ElasticSearch server. The front end written in React will allow users to search for data. Measurable outcomes are going to be the amount of time it takes to run a query against a traditional RDMS database vs. using ElasticSearch. The planned architecture is shown in Figure1:

[1] Gujarat, India, Darshita Kalyani, and Dr. Devarshi Mehta, "Paper on Searching and Indexing Using Elasticsearch," *International Journal Of Engineering And Computer Science*, June 30, 2017, https://doi.org/10.18535/ijecs/v6i6.45.



Architecture Diagram

Progress Report

The estimated time to complete this project is about 30-40 hours.

Tasks and time:

Task	Hours	Status
1. Determine how to set up, configure and deploy ElasticSearch on a cloud platform : a. The cloud platform determination will base on the cost difference between Azure, **Google** and AWS.	15 hours	**Completed.** Created an account on Google and hosted ElasticSearch as a hosted service on ElasticCloud.
2. Migration of raw to json data to Elastic search server :	3 hours	**Completed.** Created a data ingestion pipeline to read in a file which contains json per line and convert the file into an array of jsons. Then uploaded the data to ElasticSearch Cloud.

3. Development and deployment of Web application:	10 hours	**Started.** **8 hours' worth of work left.** Creating a CRUD application in REACT using the API to communicate with the ElasticSearch Engine.
4. Test and measure outcomes:	4 hours	Not started.

Issues:

Originally the plan was to develop a C#.NET application, but further research supported developing a front end using Node.js or React. I will be creating A CRUD (Create, read, update, delete) client using React.