Team Climate Change

Technological Feasibility Analysis

November 4, 2019

Project Sponsors

Doctor Geoffrey Roest

Professor Kevin Gurney

Team Faculty Mentor

Scooter Nowak

Team Members

Kiley Jacobs (Team Leader)

Mumbi Macheho-Mbuthia

Tung Nguyen

Zihang Shen

Yisheng Wang

Table of Contents

1. Introduction

The Earth is warming at an alarming rate, and the growth of CO_2 emissions is a major reason for this change. Many countries have signed agreements to reduce CO_2 emissions, and many cities in the United States, like Los Angeles, are taking the lead in reducing CO_2 emissions. Although actions like diplomatic agreements are not enough. Without an easy to use way for everyday people to see environmental changes, people cannot observe and understand the changes of regional CO_2 emissions. Also, people don't know whether their actions really mitigate global climate change or not. So, our project is to create a Web Application that can give people a very clear visual map that shows how much CO_2 emissions are increasing, as well as being able to see the CO_2 emissions level for their own hometown.

The project sponsors are Geoffrey Roest, Postdoctoral researcher in SICCS (School of Informatics, Computing and Cyber Systems) at NAU (Northern Arizona University), and Kevin Gurney, Professor in SICCS at NAU. Professor Kevin Gurney has recent research in topics in carbon cycle science, climate science, and climate science policy, and is working on the project involving simulation and quantification of U.S. CO_2 emissions, the linkages between terrestrial carbon exchange and climate variability, and the impacts of deforestation on climate. He also has worked extensively on climate policy and have been involved for over 25 years, with the United Nations Climate Change Framework Convention, and the Kyoto Protocol. For this project they have collected large amounts of data about CO_2 emissions in the United States, and have been able to create a map over the country that displays this data. The problems that they are facing right now are dissemination and accessibility. Since the data is only available in technical formats, you need specific software to open and analyze them; the general public doesn't have sufficient knowledge or skills to use most of this software. So, our primary goal is to build an easily accessible interface that allows everyday people to understand and explore CO_2 emission findings.

Our solution will be a Web application that can show the CO_2 emissions in an interactive graphical interface. Users can access our website and see a map of America displaying relative CO_2 emissions. The purpose of this document is to outline options for each major technological challenge we have identified to finish this project. We will show the pros and cons of each and choose which is the most viable for challenge. Our document will start with the different challenges we will face, followed by analyzing what we can do to face those challenges, and then discussing the methods for integrating all these separate parts into a finished project.

2. Technological Challenges

Before we dive into our analysis, we need to present an overview of our challenges. This project presents us with six challenges; Our first challenge is finding a good web framework since it's the major core in our project. Then, we need to have our own cloud environment to upload our data for the purpose of our project. This cloud must be able to take in data and allow that data to be downloaded into a database. Another challenge is to use a good database that can handle all the geographic information and spatial data. Our project also requires a map framework that can create and edit several kinds of data, visualize and interpret the data in 2D and 3D, especially raster data that our clients are using. Using a good IDE and front-end libraries will also speed up our coding and presentation, to increase project's efficiency and time resources.

2.1 Web framework

Our project requires a website that displays an interactive map; therefore, we need a robust web framework as the foundation for our web services and other technologies. Choosing the right framework from the start is very important since

this will determine what challenges can be solved efficiently and effectively later on.

2.1.1 Ease of installation

It plays a very important role while choosing a framework. A framework can pose a problem if one has to run through a number of steps just to get it installed and working.

2.1.2 Scalability

A framework has to be able to handle growing a website as well as being ported to mobile phones. Eventually our client wants the project to be able to be used on android and IOS devices.

2.1.3 Security

It is important to choose a framework which is updated often, is open about the bugs it finds, and fix them quickly. Our private data needs to be protected from web attacks, or hackers attempting to steal information.

2.1.4 Language

Our team are mostly fluent in Java and JS. This will help us save time in learning another language. Another potential language is C++.

2.2 Map framework

Since our project required to show the interactive map on a website, we need a map framework that can manage the display of our map data, handle user interactions with data and is finely tuned to handle different web browsers and devices. A suitable map framework will increase our efficiency.

2.2.1 Visualize data

The map is able to display data in 2D and 3D (if possible). It can also manage multiple maps and layouts in a project-based workflow. Also, the map should be able to zoom in and out for a large map, and display statistical data and geographic information.

2.2.2 Scalability

Map can be embedded and published within the website. The framework can maintain effective performance with increases in the growing data.

2.2.3 Edit and analyze data

Ensuring data integrity and accuracy for storing editing and managing all kinds of spatial data. The framework should be able to analyze and display the information in a statistical graph and different levels of carbon dioxide emissions.

2.3 Database

What we need for this project is a database that will serve as a storage for the raster data files. Our client has a 7 years' worth of data, which is more than 10GB, consisting of images and numerical values and they are in Raster data format. We need to find a suitable database to organize, store, and access this information easily.

2.3.1 Security

Since the database will be holding sensitive information that cannot be leaked out to the public, it is crucial that any database chosen must have security standards updated frequently.

2.3.2 Performance

The amount of data stored in a database has a great impact on its performance. The database should be able to scale and handle growing data, especially with data that gets updated frequently.

2.3.3 Transferable

There is a possibility that our client may need to redeploy their server on another system. The database should be portable to another server with ease in case the server is terminated, or our client wants to use a different one.

2.4 IDE

Choosing the right IDE is important because it will increase our productivity. Not only it can help us write code more easily, but also help us to identify bugs and errors. The IDE should at least satisfy the three requirements below.

2.4.1 Speed

Speed is important when it comes to choosing IDE. Some IDE have some serious performance issues and can take a long time to load, write code and compile projects. IDEs that tend to include fewer features are often faster.

2.4.2 Debugging

Most IDEs provide an error output window that can provide insight as to why an error occurred, but these errors often have to do with compilation as opposed to code execution. This is why debugging is so vital; it allows our team

to run the program step by step, as well as set breakpoints that halt the program in key areas.

2.4.3 Ease of use

IDEs that contain plenty of packages and libraries can become harder to use due to the increase in option and GUI elements. However this also depends on how those libraries and packages are implemented.

2.5 Front end libraries

Our website needs to have a simple to use and understand UI for users. A good front end library will help with this by allowing us to make a slicker design in a more efficient manner. Creating CSS and HTML files from scratch could easily become time consuming, so an effective library will help with this.

2.5.1 Performance

The library should be able to generate and run CSS files in a way that won't inhibit the performance of the website. It could also speed up performance by having prewritten code that is more efficient than what we would have written.

2.5.2 Good Design

The library should be easy to use and decrease the amount of time we will need to work on designing the front end. Many front end libraries are designed with people with little web programming experience in mind.

2.5.3 Compatibility

The library must be compatible with our backend web framework as well as the rest of our site; it can't use a kind of file our database or something else can't process.

2.6 Cloud

All modern websites need some form of a cloud for cross-platform data storage. Our website has no real user input or user data, such as user logins, so all cloud storage is based around the needs on our end alone.

2.6.1 Storage capabilities

Since there won't be anything stored in the cloud besides information on our end, the storage doesn't have to be as high as if we had to store user information as well. The storage only has to be able to hold the amount of data we have, which may be quite large.

2.6.2 Language

The cloud must be able to work with whatever frameworks and programming languages we use. If we decide to change what language we are writing in, the cloud should ideally support that change with causing even larger alterations.

2.6.3 Cost of Services

It will ideally be free or at a low cost as we won't need an extreme amount of storage room. A small fee we can manage if using the service is worth the cost.

3. Technological analysis

Now that we have gone through our challenges, we explored and selected the different technologies that will solve our technological challenges. In the following sections, we detail our thought process and decisions for which web, map

framework and database management software to use, as well as IDE, front-end libraries and where the information will be hosted.

3.1 Web Framework

The web framework we choose will need to have certain attributes and features which are important to us for our coming up with solution. The most important features we desire in a web framework are laid out in Figure 1 below. Some of which include: Security, language, scalability.

3.1.1 Express

Express is the most popular node web framework since it uses node.js and is the underlying library for other web frameworks. It prides itself in minimal and flexible framework. This is good for a developer who uses JavaScript, because express is fully JavaScript powered. Like Django, Express also tried to keep everything simple and powerful. Another advantage of express is it can handle many requests and notifications from users.

3.1.2 Django

Django is one of the most popular Python frameworks. Python is a language that is easy to learn with a simple design. Django markets itself as a "battery included" framework in the sense that once it is installed and configured, a developer can jump right in and start developing. Django also has built in security to prevent common attacks and SQL injections. It also inherits all Python's features to speed up website development.

3.1.3 Spring

Spring is another framework that uses Java. It provides a lightweight container that doesn't need an application server. The Spring Framework provides a comprehensive programming and configuration model for modern Java-based enterprise applications - on any kind of deployment platform. A key element of Spring is infrastructural support at the application level so that teams can focus

on application business logic. However, the learning curve is high. Spring has a vast number of classes, and many of them are encapsulated.

	Flexibility	Scalable	Language	Security
Django	Limited	Yes	Python	Good
Express	Very good	Yes	JavaScript	Good
Spring	Good	No	Java	Good

Table 1: Framework Comparison

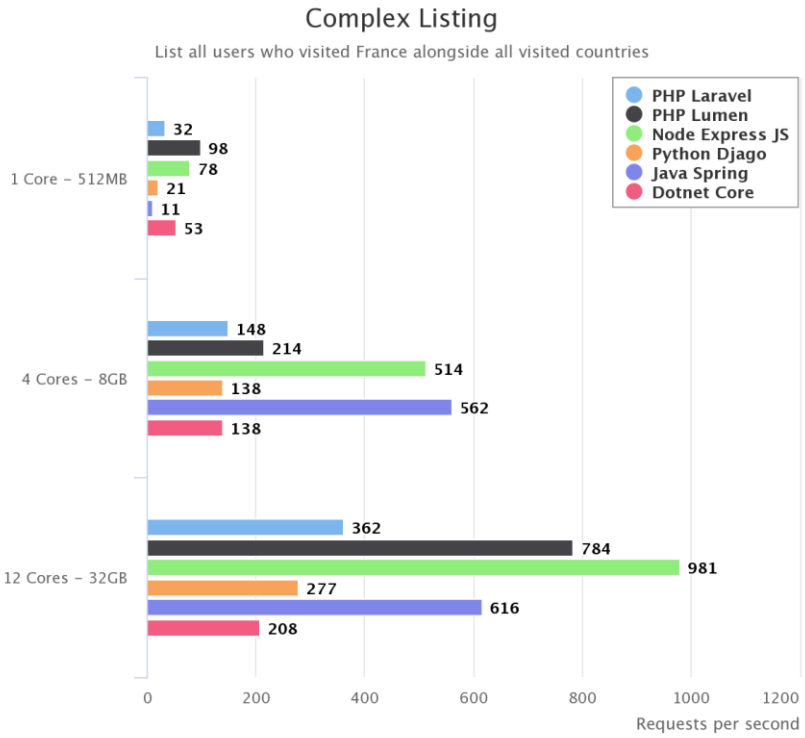

Figure 1: Benchmark comparison of several web frameworks (www.medium.com)

In Figure 1, there are several frameworks being tested based on the number of requests per seconds each framework achieves on different server configurations

alongside how clean or verbose the code looks like. They also tested on how each framework scales its performance depends on the server configuration.

Conclusion:

In figure 1, we can see that Express' results is very fast when it comes to handling big data files. It competes with technologies such as Java and .NET Core or even outperforms them and combined with the simplicity of Javascript which you can natively use with Node.js, it provides so much power. Based on table 1 and figure 1, for this project we will use Express. It is the most commonly used for Node.js development, so our team members know how to use it. It also gives us the freedom in configuration and setup. Furthermore, it integrated many useful tools and functionalities that will save us time while developing the website. Express's robust API allows us to configure routes to send/receive requests between the front end and the database.

3.2 Map Framework

For the framework of our map, we have a few challenges that need to face. It needs to be an interactive map, with scalability, update frequently, and use Gps for users' locations. Now, we have four options: Mapbox GL JS, Polymaps, OpenLayers, and ArcGIS integrate with raster data manager.

3.2.1 OpenLayers

In the framework of OpenLayers: the whole map is regarded as a container (Map), the core is the map layer (Layer), each layer has a corresponding data source (Source), and the map view (View) for map performance. The map container also supports some controls (Control and Interaction) that interact with the user. In addition, OpenLayers also supports event mechanisms.

3.2.2 Mapbox GL JS

The Mapbox GL JS is a very common map framework. It is a JavaScript library that uses WebGL to show the interactive maps from vector and raster data. Our data are based on raster and most of the members in our team prefer to use JavaScript. Also, the CSS can be used in Mapbox GL JS to style DOM elements. Mapbox GL JS is really fit for our project. The CSP directives used in Mapbox GL JS can help mitigate web security vulnerabilities to specific security policies. The Mapbox API can let us change the map styles, fonts, and images as can be seen in figure 2. It also provides some good style; we may use it to reduce our coding time.

Figure 2: Style that Mapbox provides

3.2.3 Polymaps

Polymaps was very similar to Mapbox GL JS. Its target language is also JavaScript and works for Web development. Vector and raster data also can work in Polymaps. The advantage for Polymaps is it can load data on a large range of scales, it fits our goal to show the U.S. map. SVG (Scalable Vector Graphics) in Polymaps can let the user have a very nice interaction within scalable maps.

3.2.4 ArcGIS

ArcGIS is built around the Geodatabase database and use Python,JS, C# or C++ to written in. As part of the Esri Geospatial Cloud, ArcGIS Online connects people, locations, and data through interactive maps. Intelligent data-driven styles and intuitive analysis tools that provide location intelligence. The unique immersive experience of interactive maps transforms maps from static views to opportunities for user exploration. Get additional details by zooming in, view information about a specific area as text, tables, and images by clicking on the map, search for world locations and your custom location, and other perspectives by filtering data and changing colors.

	Website	Language	Interactive	Scalability	Raster	IOS & Android`
Mapbox GL JS	Yes	JS	Yes	Good	Yes	Mapbox SDK for Xamarin
Polymaps	Yes	JS	Yes	Good	Yes	No
OpenLayers	Yes	JS	Yes	Good	Yes	No
ArcMap/ArcGIS	Yes	Python, JS, C#, and C++	very good	very good	yes	Yes

Table 2: Map framework comparison

Conclusion:

Based on the table 2 results, all these four frameworks are matched to our project challenges and requirements. However, the Mapbox has other version for IOS and Android, so if we want our website to work for IOS and Android, Mapbox is a good choice. And since the ArcGis is very good on interactive and scalability, it can be our second choice.

3.3 Database

And next is our most important part, the map. Our map resources include data for every city from 2010 to 2015, it's a big database resource. Now we have found three suitable database server options for our project, and discuss them here.

3.3.1 MongoDB

MongoDB is documented-oriented NoSQL database used for high volume data storage. It can store any data type and doesn't require schema. The MongoDB is scalable horizontally with millions of documents within the database. It offers flexibility with JSON documents which means data structure can be changed over time.

3.3.2 Rasdaman

Rasdaman (raster data manager) is an Array DBMS, that is: a Database Management System which adds capabilities for storage and retrieval of massive multi-dimensional arrays, such as sensor, image, simulation, and statistical data. A frequently used synonym to arrays is raster data, such as in 2-Draster graphics; this actually has motivated the name Rasdaman. It is suitable for a database, picture resource like map. We will have a lot of modifications on our map

resource, Raster data manager is useful in modifying, creating, and deleting database sources.

3.3.3 Geodatabase

Geodatabase is a data model that uses standard relational database technology to represent geographic information. Geodatabase supports the storage and management of geographic information in standard database management system (DBMS) tables Geodatabase supports a variety of DBMS structures and multi-user access, and its size is scalable. Geodatabase supports everything from small single user databases based on Microsoft jet engine to multi-user databases at workgroup, departmental and enterprise levels.

	MongoDB	Rasdaman	Geodatabase
Query Language	NoSQL	RaSQL	GIS object model
Flexibility	High	High	High
Handle spatial data	Medium	High	High

Table 3: Database Comparison

Conclusion:

Looking at the comparison in table 3, we choose Rasdaman for our database. The main reason is Rasdaman is made to handle spatial data and provides

flexible, fast geo services for multi-dimension image and statistical data. Our second option is Geodatabase since it optimized for use with ArcGIS.

3.4 IDE

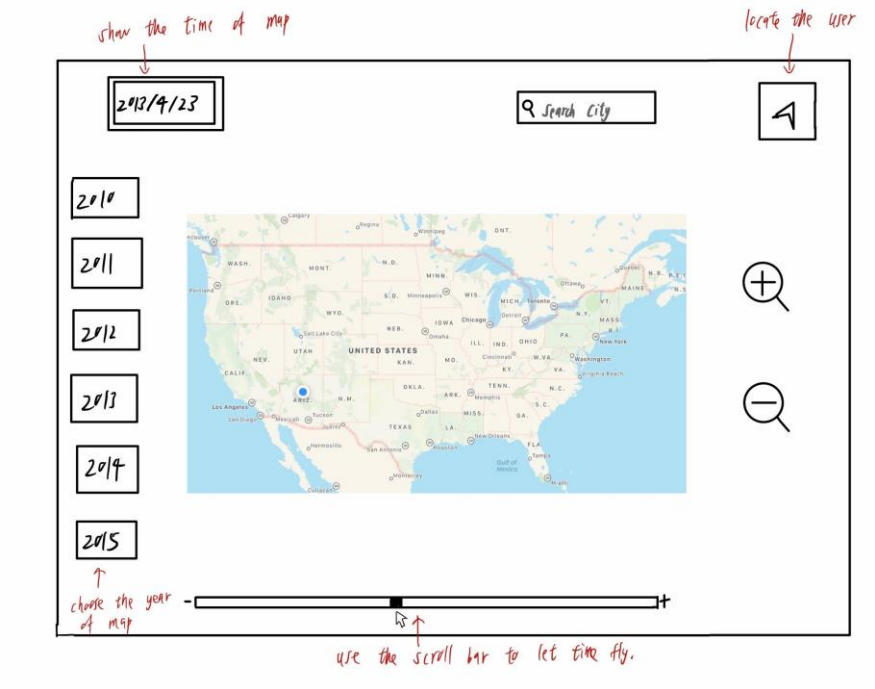

Figure 3: Map layout

Our map layout applies many common functions in a map website, for example, the yearly button in the left means showing our map for each year. When users scroll the map to the bottom, they can see CO_2 emissions displayed in different cities at different times. The scroll bar also shows CO_2 emissions in the USA from January to December. The time of the map is shown on the top left corner. In addition ,users can also enlarge or narrow the size of the map, and use the location button to locate their positions or directly search any city on the map.

Now we have designed a draft for our website, the picture above shows what our website will look like. Our first step is to finish the front-end part of the website, which requires CSS, JS, and HTML, we found three front-end IDEs and compared them.

3.4.1 Webstorm:

Webstorm is a JavaScript development tool forJetBrains. It's very useful in writing CSS and HTML, which is also compatible with JavaScript and homologous with IntelliJ idea, and inherits the powerful JS part of IntelliJ idea. Our project mostly focuses on HTML and CSS, so I think this software will play an important role in our website.

3.4.2 Eclipse

Eclipse is an open source, Java based, and extensible development platform. In itself, it is just a framework and a set of services for building development environments through plug-in components. It's very useful in writing Java, which is always used in writing PHP and Python, we may use php in our website, so Eclipse will take part in our project.

3.4.3 Visual studio

Microsoft Visual Studio (VS) is a series of development kit products of Microsoft Corporation. VS is a basic and complete development tool set, which includes most of the tools needed in the whole software life cycle, such as UML tools, code control tools, integrated development environment (IDE), etc. The target code is applicable to all platforms supported by Microsoft. We use it in writing SQL and link database.

	Webstorm	Eclipse	Visual Studio

Performance	Fast	Fast	Normal
Language	JavaScript	Java	Many
Ease of use	Good	Good	Good

Table 4: IDE Comparison

Conclusion:

As can be seen in table 4, Webstorm and Eclipse perform faster than Visual Studio. Since our project is about developing website, we choose Webstorm for our IDE. Webstorm also provides advanced coding assistance for node.js.

3.5 Front-end libraries

When choosing our front-end library what matters most is that it is easy to use, simplifies the front end design process, and is compatible with our project. We have looked at and compared two popular front-end library tools.

3.5.1 Bootstrap

Bootstrap is a commonly used front end library. It is known for coming with starter templates. It is used for both CSS and Javascript. It does not include jQuery though. This is the library our team has the most experience with

3.5.2 Bulma

Bulma is seen as the most popular alternative to Bootstrap. It is very simple to install and more friendly for beginners. However, it is much more CSS and HTML based and is not javascript focused.

	Bootstrap	Bulma
Resources	Many	Plentiful
Install ease	moderate	very simple
Javascript	yes	no
jQuery	no	yes
Experience	some	none

Table 5: Front-end Library Comparison

Conclusion:

We are going to use Bootstrap because we have team members who have experience with it, and it also is more compatible with javascript. Bulma may be easier to use, but Bootstrap is more viable for us.

3.6 Cloud

What we need for this project is a cloud that we can upload and download from that will serve as storage for our data files. Preferably, this cloud will have supported Java integration and implementation to allow consistency of the project and allow a free to use solution for the duration of the project. There are two options that we consider: Amazon Web Services and Google Cloud Platform.

3.6.1 Amazon Web Services

Amazon Web Services (AWS) is a secure cloud service platform that lets you access your files anywhere. AWS also lets you run web and application servers in

the cloud to host dynamic websites. Another useful feature of AWS is you can manage databases such as mySQL or SQL to store information.

3.6.2 Google Cloud Platform

Google offers a secure and durable cloud storage that allows instant access to data anywhere. They can integrate storage into your app with a single API. They offer a 1-year trial access to all Google Cloud products with $300 credit for new users to spend on the Google Cloud Platform. For free users, the limit is 5GB of storage a month.

	Amazon Web Service	Google Cloud Platform
Language	Many	Many
Storage	5GB	5GB
Learning Curve	High	Medium
Cost	Free 1 year	Free 1 year

Table 6: Cloud Comparison

Conclusion:

According to table 6, both Amazon and Google offer very good cloud storage. However, we chose Google because it has step by step examples of how to do many common tasks, which is very useful for first time users. In addition, Google is a large company that has one of the highest ratings for security.

4. Technological Integration

We now have to consider how to integrate all these components to complete the requirements for this project.

When the data is downloaded from the cloud, we have to evaluate those data and eliminate incorrect data. This can be done with a backend program written in Java and then display that data on the map.

Figure 4 is a basic representation of how we want to integrate our technologies. Our data will be stored in Google Cloud Platform and downloaded to Rasdaman. MapboxGl will use organized data from Rasdaman to display an interactive map of United State that shows CO_2 emissions statistic data and other geographical information when a user hovered over. The Map will be embedded on our website. The website will be implemented in standard HTML, CSS, JavaScript using Express Framework.

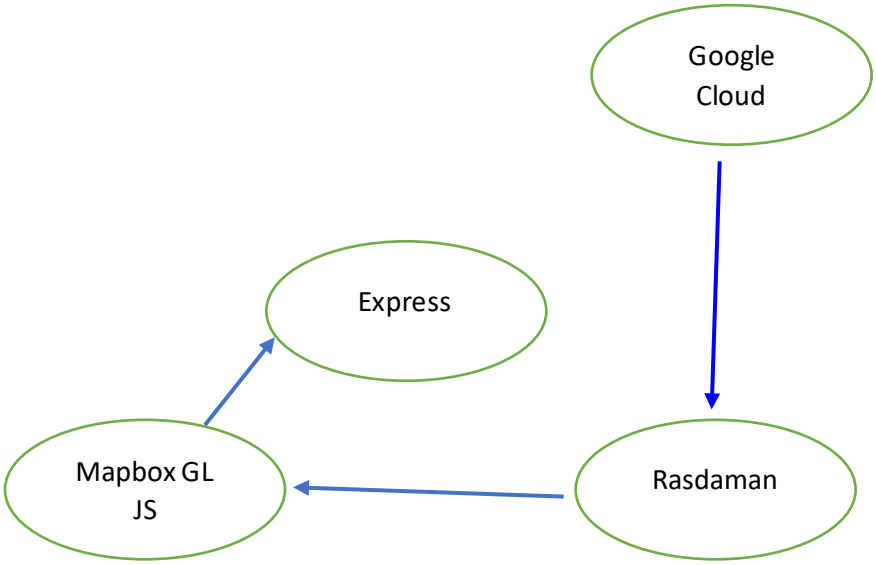

Figure 4: System Diagram

Our website will be hosted on the Google Cloud Platform with Firebase hosting. We will have a free 1-year trial which will last for this project. Since our data is stored on the files, we have to use our backend program to transfer that data to the database on google cloud. The application will have to be able to upload the data to the database as it being used. It must also be able to read the database in real time. The

application must also be able to use data from Rasdaman to display statistics on map. Our framework Express will help create server-side web apps faster and more optimized.

5. Conclusion

The project given to us by our client is to help create a web application that people can use to see the CO2 emissions in the United States that doesn't require technical knowledge of the data. So, our solution to this is to interpret this data in a database that will allow the website visitor to view the data in a number of ways simply by clicking buttons and sliders. Our current design on the interface is in figure 3.

This documents purpose was to go in depth with the tools we will need to accomplish this goal by analysing different programs to use to develop all the different parts. Below is a table of the choices we made for each development tool.

Tool needed	Chosen Option
Web Framework	Express
Database	Rasdaman
Map	Mapbox GL JS
Front-end Library	Bootstrap
Cloud	Google Cloud Platform
IDE	Webstorm

Table 7: Final Tool Choices

Moving forward, the more we use these tools, the more we will see how realistically feasible they are to use, so we are open to these choices changing. The main thing

we need to work on is seeing how well our integration plan will work, as with so many parts it is easy to have difficulty with this aspect.

Currently, we have a good plan moving forward for fully diving into development. For several of the tools we picked, we have team members who already have experience with them, which will mean we will spend less time getting used to them. We also tried to choose technologies that have built in parts to make creating our website easier and simpler, such as premade CSS blueprints, which will also make development take less time. Overall, we feel we will have minimal issues with the tools we use to make our application and can rather focus our efforts on problems related to the actual content of our project.