

Capstone project report

Dandan Liu

Contents

Abstract.....	4
1 Introduction.....	5
1.1 The history of the website	5
1.2 How web browser works.....	6
1.3 The website and data analysis are useful today	8
2 Web developments	9
2.1 Front end development.....	10
2.1.1 Hyper Text Markup Language (HTML)	10
2.1.2 Cascading Style Sheet (CSS).....	11
2.1.3 JavaScript	16
2.2 Back end development and database	19
2.2.1 Node.js.....	20
2.2.2 Express.js.....	21
2.2.3 Database	21
3 Data analysis	22
3.1 Data loading	22
3.2 Data visualization	26
3.3 Data modelling	27
4 Website deployments.....	28
5 Concluding remarks.....	34
Reference	35

Tables

Figure 1 Asking for IP address.....	7
Figure 2 Getting the correct web page.....	7
Figure 3 The website is working	8
Figure 4 Web development	9
Figure 5 Website using HTML5	11
Figure 6 CSS properties.....	12
Figure 7 Colour board.....	13
Figure 8 Fonts property	14
Figure 9 Home page	15
Figure 10 Using HTML and CSS.....	15
Figure 11 Register page	18
Figure 12 How back end works.....	19
Figure 13 Back end libraries review.....	21
Figure 14 Data loading	23
Figure 15 Statistic summary	24
Figure 16 Data frame.....	26
Figure 17 Graphs	27
Figure 18 Different languages for deployment	28
Figure 19 The problem that I had.....	30
Figure 20 How to choose database (a).....	30
Figure 21 How to choose database (b)	31
Figure 22 Install database for the project.....	32
Figure 23 Create own database	33
Figure 24 Available database.....	33

Abstract

This project presents a website which is also a web application for data visualization and machine learning. This project contains three parts which are front-end website development (mainly focusing on HTML5, CSS3, bootstrap and JavaScript), back-end server (mainly working with database and focusing on Node.js and Express.js) and deploying and launching the web application securely.

In the first part, I built the text frame which includes title, header, body and footer for the website using HTML5. After that, in order to make the website beautiful and interactive, I used CSS to add some styles like background colour, layout, fonts and used JavaScript to create some actions and behaviors so that people can subscribe or register if they are interested in the website.

In the second part, I built the data analysis which includes data loading, data visualization and data modelling. First a python flask script was written to connect the graphing and modeling functionalities with the (front-end) HTML web page. The graphing and modeling were implemented using Python matplotlib library and scikit-learn library, respectively, which are well-known and powerful data analysis tools. The buttons on the web page can trigger the Python part to conduct any analysis and then send the analysis result to the web page and display.

In the third part, I looked for servers to host this web application. Since the website has data analysis functionality, I did not choose something like HostGator to host the website. HostGator are really good at hosting and serving simple HTML projects which just have some text information on the website. There are some options to host web applications like Amazon web service, Digital Ocean, Engine Yard, google, Heroku and so on. I chose Heroku because it is really developer friendly and they offer a free tier which is good for my project.

Nowadays, we are living in a big data world, and lots of data are being collected from different industries. How to use those data to get the useful information is becoming more and more important. Loading data to the website, and with just a few buttons it can show different kinds of graphs which might provide some

information hidden in the data. If the user has historical data and wants to make predictions for the future, the website offers lots of modeling methods which can be used to train models and use the model to predict the future information.

1 Introduction

1.1 The history of the website

Before the website exists, the internet already grew fast. Since each computer has its own way of doing things, it is hard for computers to communicate with each other like to share the academic papers and so on. They need a way to share documents so the World Wide Web ("WWW" or simply the "Web") was developed in 1990 by Tim Berners-Lee for the NeXT Computer (at the same time as the first web server for the same machine) and introduced to his colleagues at CERN in March 1991. There are three core technologies of the World Wide Web content engineering, which are HTML, CSS and JavaScript.

HTML, which stands for Hypertext Markup Language, was first proposed by Berners-Lee in 1989. In 1980, physicist Tim Berners-Lee, a contractor at CERN, proposed and prototyped ENQUIRE, a system for CERN researchers to use and share documents. In 1989, Berners-Lee proposed an Internet-based hypertext system. Berners-Lee also wrote the browser and server software in late 1990. He is the first web developer.

CSS, which stands for Cascading Style Sheets, was first proposed by Haakon Wuisman Lie on October 10, 1994. At the time, Lie was working with Tim Berners-Lee at CERN. Several other style sheet languages for the web were proposed around the same time, and discussions on public mailing lists and inside World Wide Web Consortium resulted in the first CSS Recommendation (CSS1) being released in 1996. Style sheets have existed in one form or another since the beginnings of Standard Generalized Markup Language (SGML) in the 1980s, and CSS was developed to provide style sheets for the web. One requirement for a web style sheet language was for style sheets to come from different sources on the web. Therefore, existing style sheet languages like DSSSL and FOSI were not suitable. CSS, on the

other hand, let a document's style be influenced by multiple style sheets by way of "cascading" styles.

JavaScript was created in 1995 by the Netscape web browser in order to add actions to their website. At that time the other browsers started adopting their own version of JavaScript. Each browser has their own JavaScript. Eventually a standard had to be created to make sure that web developers can build one website and have it work on all browsers. The standard is called ECMAScript. Usually we call it JavaScript as well. JavaScript is a high-level, and interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm. It is used to make web pages interactive and provide online programs, including video games. The majority of websites employ it, and all modern web browsers support it without the need for plug-ins by means of a built-in JavaScript engine. Each of the many JavaScript engines represents a different implementation of JavaScript, all based on the ECMAScript specification, with some engines not supporting the spec fully, and with many engines supporting additional features beyond ECMA. As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has an API for working with text, arrays, dates, regular expressions, and basic manipulation of the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded.

1.2 How web browser works

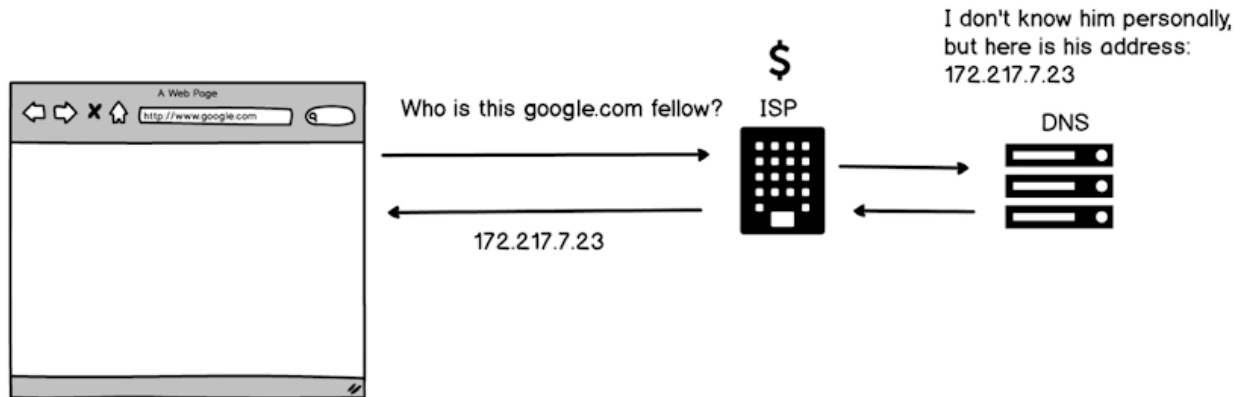


Figure 1 Asking for IP address

In Figure 1 Asking for IP address, when we want to visit a website like google, we type in the URL (stands for Uniform Resource Locator), then the message goes to ISP (stands for internet service provider), finally it goes to DNS (stands for domain name server) who has the list of the URL and the corresponding IP address. The IP address gets back to the browser.

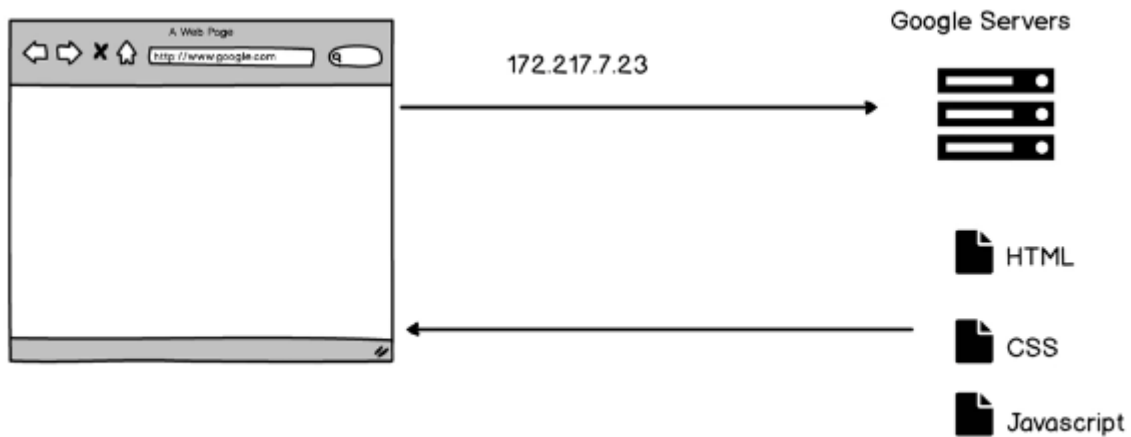


Figure 2 Getting the correct web page

In Figure 2 Getting the correct web page, when the web browser gets the IP address, it sends another request based on the IP address. Based on the IP address, the request goes to the server who hosts the files like HTML, CSS, and JavaScript for the web page. The server will send the HTML file first then the browser starts to read the HTML file line by line. When it needs the CSS or JavaScript, it will send the request to the server again to ask for those files. Finally, all the files get sent to the browser from the server then the web page is showing as we expected.

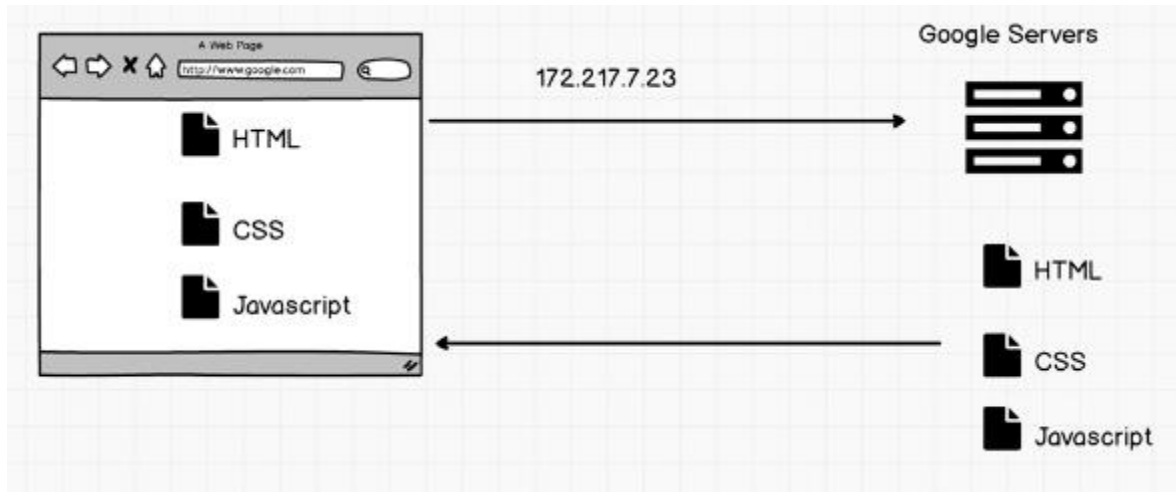


Figure 3 The website is working

In Figure 3 The website is working , based on how the website works, if we want to improve the browsing speed, there are a few things that we can do. First, we can relocate the server to make sure it is close to all the users. Second, developers should try their best to reduce the number of requests to the server for the page. In this way it reduces the trips so it will be faster. Last but not least, we should reduce the size of the file so it makes the travel time shorter to improve the performance.

1.3 The website and data analysis are useful today

Nowadays, lots of people like some youtubers or bloggers have their own websites to share their life styles or fashion, and more and more people choose to open their own small firms. The demand for website is increasing dramatically. At the same time, big data and machine learning is the new trend. The benefits that big data

analytics brings to the table, however, are speed and efficiency. Whereas a business would have gathered information, run analytics and unearthed information that could be used for future decisions or business can identify insights for immediate decisions. The website with data analysis is very useful today.

2 Web developments

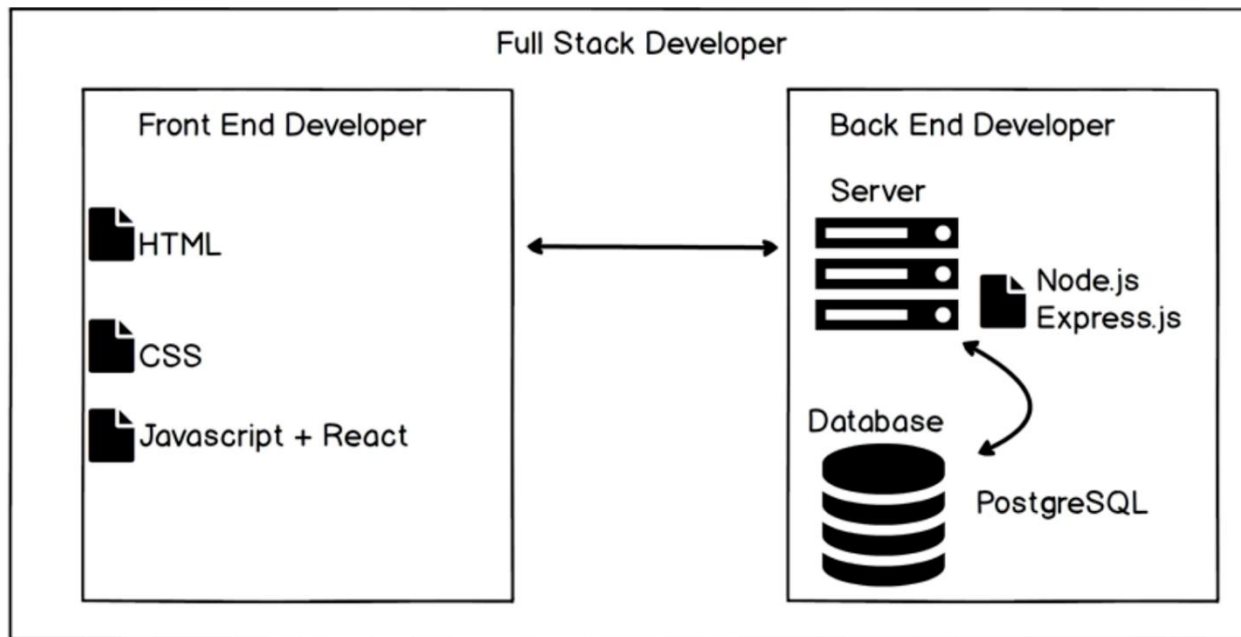


Figure 4 Web development

As shown above in Figure 4 Web development, there are 2 parts for web development, which are front end and back end which is also going to talk with database. Front end is mainly working on HTML, CSS, JavaScript, and some libraries like Bootstrap, jQuery, React and so on, in order to make sure that when users open up the browser and type in the URL, they can get the nice and correct layout page they are looking for. Back end is mainly working on the server using Python, Node.js, Express.js, or some other languages, in order to make sure that when the browser speaks to the server, the server is up and the file the server needs is ready. The server is also going to communicate with the database to get the data it needs and to store the login table and subscribe table.

2.1 Front end development

Front end is mainly responsible for making sure everything the user can see is correct like the suitable fonts, the adjustable layout based on different screen size and so on. There are 3 parts for front end: HTML, CSS and JavaScript.

2.1.1 Hyper Text Markup Language (HTML)

HTML describes the structure of Web pages using markup. HTML elements are the building blocks of HTML pages and represented by tags. HTML tags label pieces of content such as "heading", "paragraph", "table", and so on. Browsers do not display the HTML tags but use them to render the content of the page.

There is different revolution of HTML. From 1991 to 1999, HTML developed from version 1 to version 4 and in 2008, the first HTML5 public draft was released. There are lots of new elements in HTML5 like semantic elements, attributes of form elements, graphic elements and so on.

I mainly learned HTML5, which is just files that each browser can read properly and figure out what the website looks like. I learned a lot from the educational website W3SCHOOL. In order to become familiar with those tags, I just played around to see what the difference is by choosing different tags. There are two different types of tags: regular tag and self-closing tag. In HTML, there are some tag that must have the close tag as well like ``, ``, `<a>`. `</+the tag>` means to close this tag for this kind of tag open and close must be paired. There are also some other tags that are self-closing tags which means they do not need the close tag like `
` and ``. The line break tag and image tag do not need the close tag. HTML also has inline element like `` and block element like `<div></div>`. To be familiar with something like if we want to change the text layout of the website where I should change on the HTML files. Also, we can use the anchor tag `` to make sure all the web pages are linked with each other and when we click corresponding buttons it can link to the page we want. In HTML5, there is also feature that the web page layout can be adjustable based on the screen size. Using HTML5, I initially built a website like Figure 5 Website using HTML5.

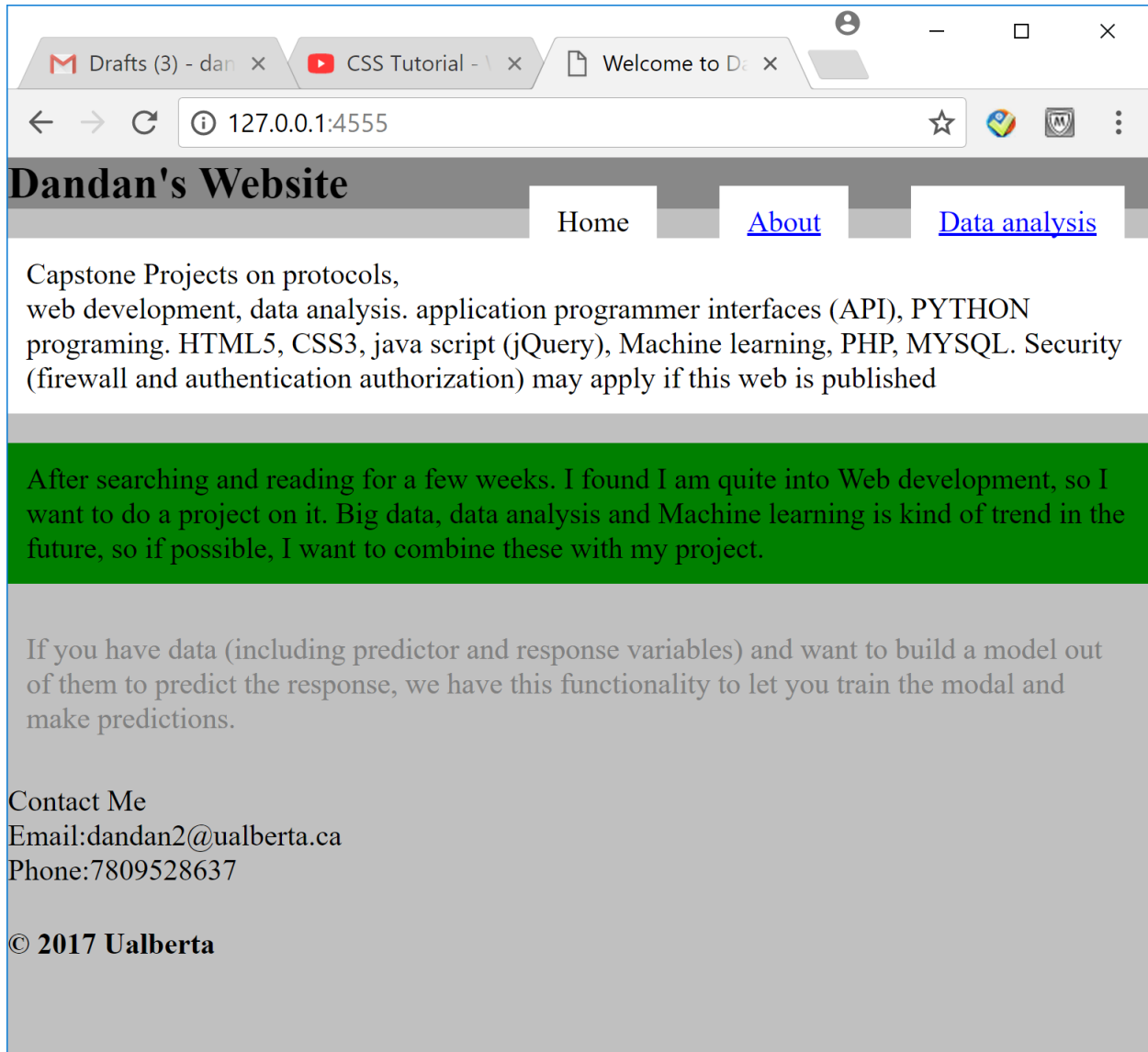


Figure 5 Website using HTML5

2.1.2 Cascading Style Sheet (CSS)

CSS describes how HTML elements are to be displayed on screen, paper, or in other media. CSS saves a lot of work. It can control the layout of multiple web pages all at once.

I learned some CSS in order to make my website looks better by adding some fonts, adding some background colour, and adding some padding or margins. When we want to write a CSS file, the first thing we want to do is to select a tag or class or id from the HTML file and then put the property and the value in curly brackets. For example, `h2{colour: red;}`. So, in CSS the main thing is to learn those properties and those selectors. I used the website CSS-TRICKS to learn those. I went to the website and then to CSS Almanac and saw they list lots of CSS selectors and CSS properties like in Figure 6 CSS properties.

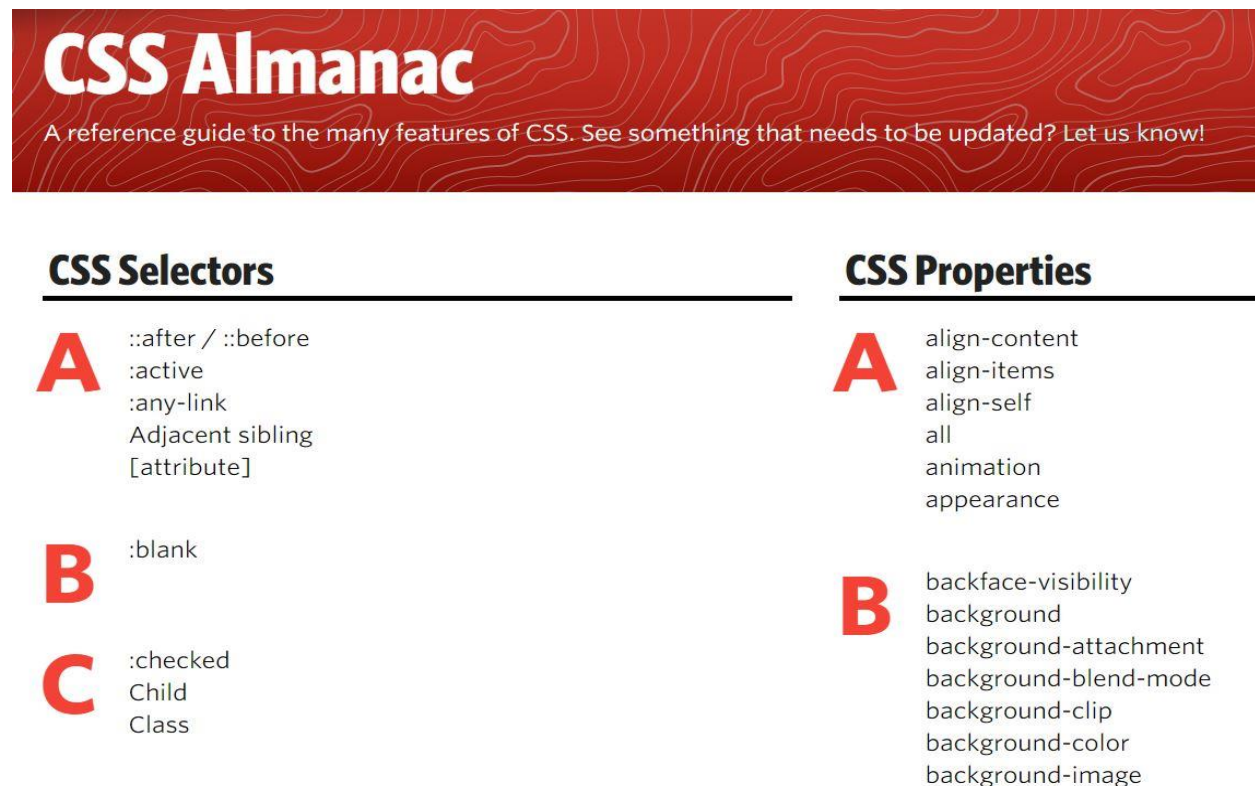


Figure 6 CSS properties

I looked for which property I wanted to learn and clicked that property so it linked me to the page which has the introduction and some examples for people to learn and play around with. I wanted to be familiar with those properties. I spent some time playing with this website and in the future if we see some new properties we can just go to this website and learn. There is no way a web developer can remember

all of them. We just need to know and remember those properties which are in use quite often, and for the other properties when we see them we need to know where we can find them and learn them fast. There is also a useful website to pick colours which is called “paletton.com”. There are RGB numbers associated with each colour and we can just put the #+the RGB hex number or the RGB number as the value. For example, I like the colour in Figure 7 Colour board. I wrote `h2{background colour: # 79b694;}` or `h2{background colour: 121, 182, 148}`.

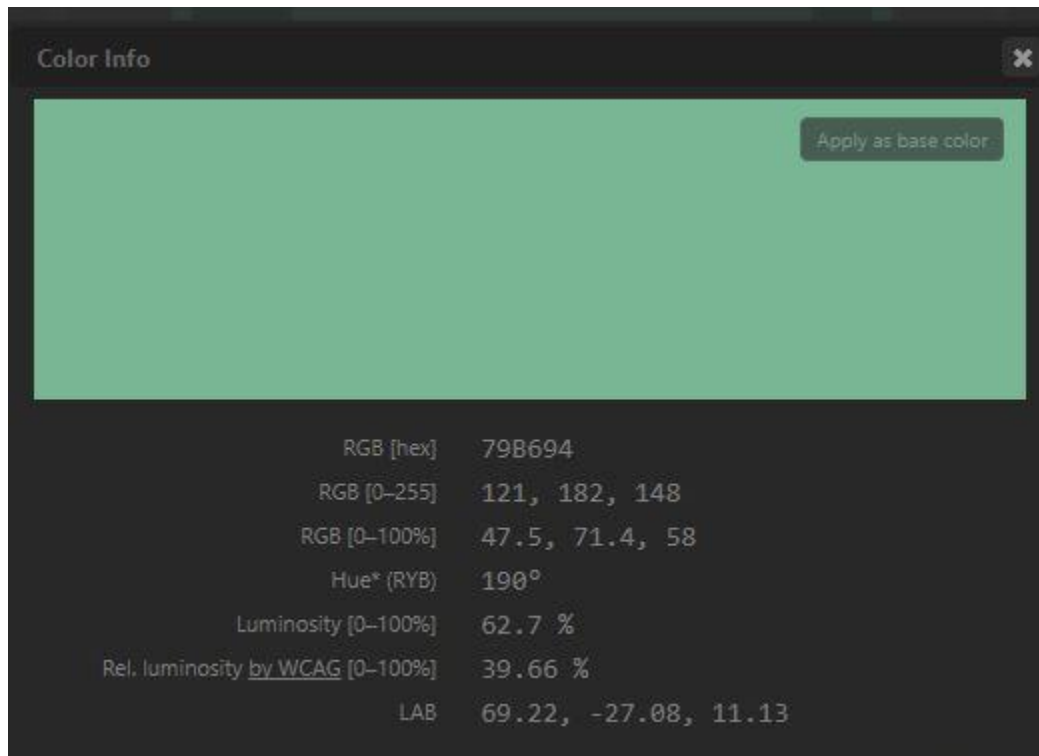


Figure 7 Colour board

There is also a good website to choose font which is “google font”. People can search and find the font they like. First, we need to add the link to the HTML files and then copy the CSS command to the CSS files. For example, I like the font which has the property of Figure 8 Fonts property. I put `<link href="https://fonts.googleapis.com/css?family=Dancing+Script" rel="stylesheet">` in the HTML files and put `font-family: 'Dancing Script', cursive;` in the CSS files.

1 Family Selected

Your Selection [Clear All](#)

Dancing Script

EMBED

CUSTOMIZE

Load Time: Fast

Embed Font

To embed your selected fonts into a webpage, copy this code into the <head> of your HTML document.

STANDARD **@IMPORT**

```
<link href="https://fonts.googleapis.com/css?family=Dancing+Script" rel="stylesheet">
```

Specify in CSS

Use the following CSS rules to specify these families:

```
font-family: 'Dancing Script', cursive;
```

For examples of how fonts can be added to webpages, see the [getting started guide](#).

Figure 8 Fonts property

With HTML and CSS, I built the website as in Figure 9 Home page and Figure 10 Using HTML and CSS, which is adjustable on screen size.

DANDAN'S WEBPAGE

[Home](#) [About](#) [Data analysis](#) [Contact](#)



My hometown

Jilin is located in northpart of China. Its the only city in China have the same name as the province.

[Read More](#)



My university

Jiangnan University is located in the middle of China. In Wuxi city Jiang su province.

[Read More](#)



My university

University of alberta.

[Read More](#)

Figure 9 Home page

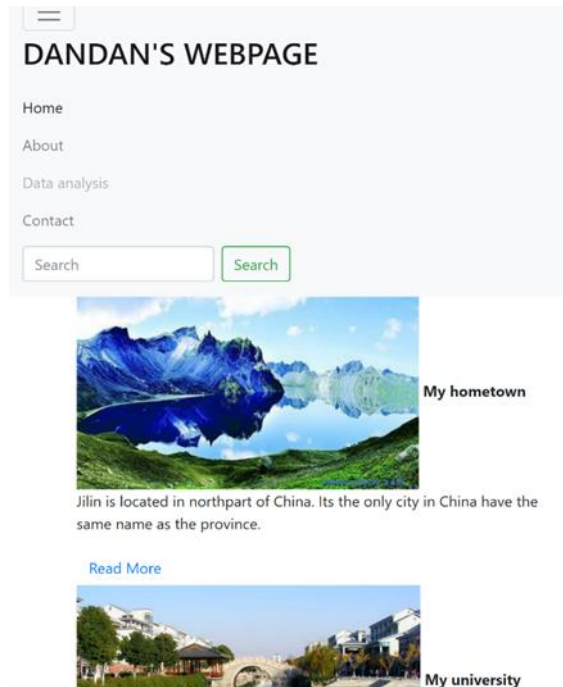


Figure 10 Using HTML and CSS

2.1.3 JavaScript

HTML is to define the content of web pages, and CSS is to specify the layout of web pages and JavaScript is to program the behavior of web pages. JavaScript are able to add actions to websites. It is like a verb in anything we do on the websites like playing the video or signing up for the user name.

JavaScript is like other programming languages, it has its own types, comparisons, variables, conditionals and so on. It has 7 types, which is the values that JavaScript can have, like number, string, Boolean, undefined, null, symbol and project. It also has some comparisons like “! ==”, “===”, “>=” and so on. JavaScript variables is where the program can memorize things. In JavaScript the variable can not start with the symbols and we need to camel case the variables, which means if there are two words for a variable we need to capitalize the first letter.

In JavaScript we use the conditional execution (i.e., if, else, else if, ternary operator or switch) to do the flow control. Through command as the following we can execute different parts of the code based on the conditions.

if, else if, else formula:

```
if (condition1) {  
    block of code to be executed if condition1 is true  
} else if (condition2) {  
    block of code to be executed if the condition1 is false and condition2 is true  
} else {  
    block of code to be executed if the condition1 is false and condition2 is false  
}
```

switch formula:


```

switch(expression) {
  case n:
    code block
    break;
  case n:
    code block
    break;
  default:
    code block
}

```

After flow control I learned loops (i.e., for loop, while loop, do while loop and so on) in JavaScript.

for loop formula:

```

for (statement 1; statement 2; statement 3) {
  code block to be executed
}

```

Statement 1 is executed before the loop (the code block) starts.

Statement 2 defines the condition for running the loop (the code block).

Statement 3 is executed each time after the loop (the code block) has been executed.

while loop formula:

```

while (condition) {
  code block to be executed
}

```

do while loop formula:

```

do {
  code block to be executed
}
while (condition)

```

The difference between while loop and do while loop is while loop will justify the condition if the condition is true it will be executing the code but do while loop will run the code first then justify the condition. If the original condition is false while loop will be not executing any code block but do while loop will be executing the code once.

After those I learned JavaScript functions. A JavaScript function is a block of code designed to perform a particular task. A JavaScript function is executed when "something" invokes it (calls it).

function formula:

```
function the name of the function (p1, p2) {  
    return p1 * p2; // The function returns the product of p1 and p2  
}
```

If we want to run this function we need to write the code “the name of the function ()” which means to execute the function.

Using JavaScript, I built a page where the user can subscribe or register with their email as in Figure 11 Register page.

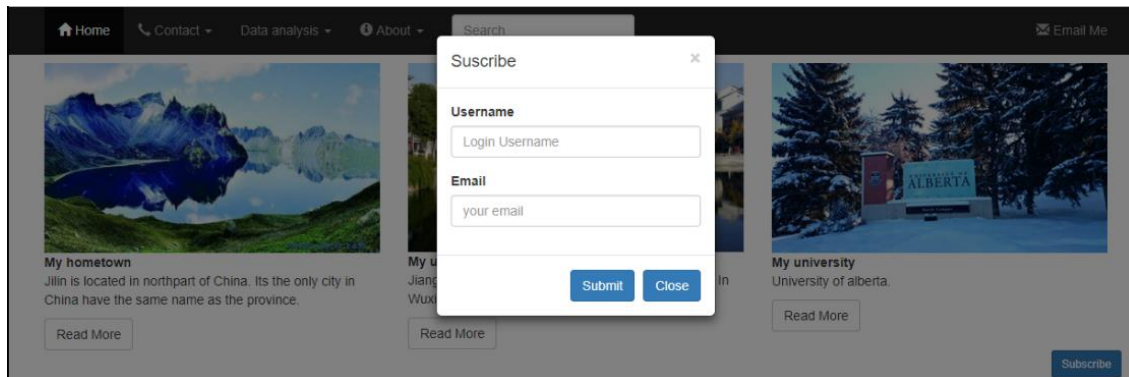


Figure 11 Register page

2.2 Back end development and database

Back end is in charge of servers and databases. Server is responsible for whatever people get in the browser. The browser makes the request and the server will response to the things that the browser asked for. It is like in Figure 12 How back end works.

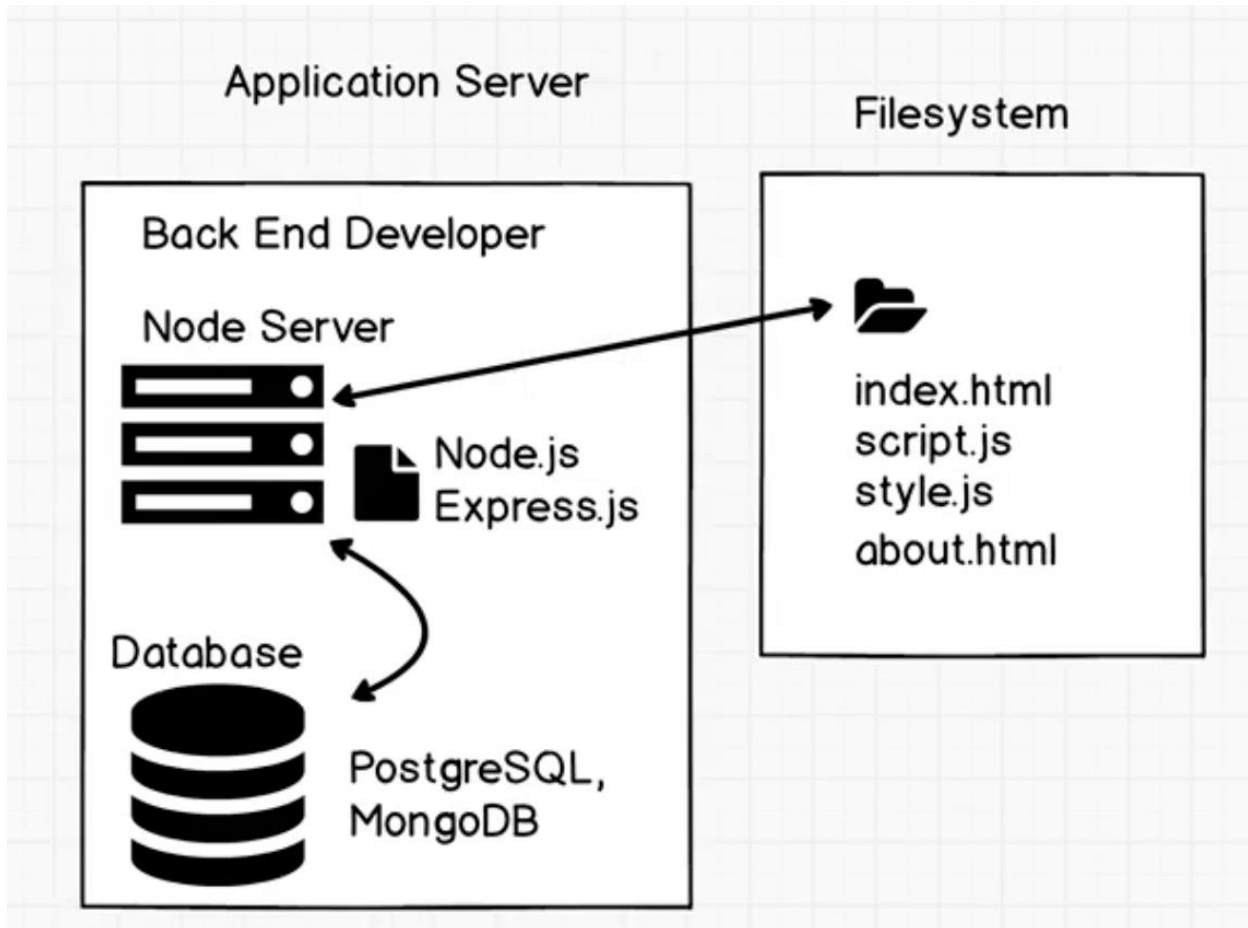


Figure 12 How back end works

For example, in the Node server, we have node.js and express.js running. We can write JavaScript if we receive a request for a login then check the user's username and password in the database. If matching, then let the server know, and the server will send a response through Jason to the front-end application. In the Node server,

we can customize what the JavaScript says, which means we can use JavaScript at the back end as well.

2.2.1 Node.js

Before Node.js was created, JavaScript was only allowed to run at front end since the browser has something like V8 engine which can read the JavaScript file. With Node.js, JavaScript can run anywhere like computer robots as long as they have Node.js.

There are 3 modules that we can use in node. The first one is the modules which we can create by ourselves. It has its module exports and we give it the path to the file so we can find it. They also have something called built-in modules which are already pre-installed with node. Like the FS (standing for file system) module which has lots of things and it allows us to read the file system and extract how many times a word appears in one file. Also, there is a HTTP module which allows us to build the server with it. The third module is from NPM. It is a package like Nodemon. A package is listening to the console, and if anything is wrong or something happens it will keep updating it.

We can use the http module in Node.js to easily build the simple server using the following command:

```
Const http = require('http');
Const server = http.createServer((request, response)
{
Response.setHeader('content-type','text/html');
Response.end('the information to show in the web page.')
}
Server.listen(3000)
```

In this way, we can open up the browser and go to the 'localhost:3000'. We will get the response.

2.2.2 Express.js

There are also many libraries that people use with Node.js to create a server, based on Figure 13 Back end libraries review. The most popular one is Express.js now.



Figure 13 Back end libraries review

Using express, we can simply build our server with a few command lines like:

```
Const express = request('express');  
Const app = express();  
App.listen(3000);
```

2.2.3 Database

All companies store data. Now in order to store this information, every company and sometimes individuals need something called database. Database is a collection of data which includes numbers, dates, passwords, and so on, and allows us to organize the data in a way that is useful to us. Database also makes data management easy. There is something called database management system. It is a collection of

programs which allows us to access databases and work with data and it also allows control access to database users.

There are different kinds of database like relational databases where the schema or those tables have to be pre-defined like MySQL, PostgreSQL and non-relational database like mongo DB.

3 Data analysis

Nowadays we all live in the big data world. We collect data from every field, and then how to get the information we want from those data is very important. Therefore, this project offers free data visualization and simple modelling parts. You just import the data and press a few buttons, and you can get the corresponding graphs. From there, you can get some statistics like the maximum, minimum, average, standard deviation and so on. There are mainly two parts in my data analysis which is data visualization and data modelling.

3.1 Data loading

There is a button on this data analysis page where user can load in the data like

Figure 14 Data loading .

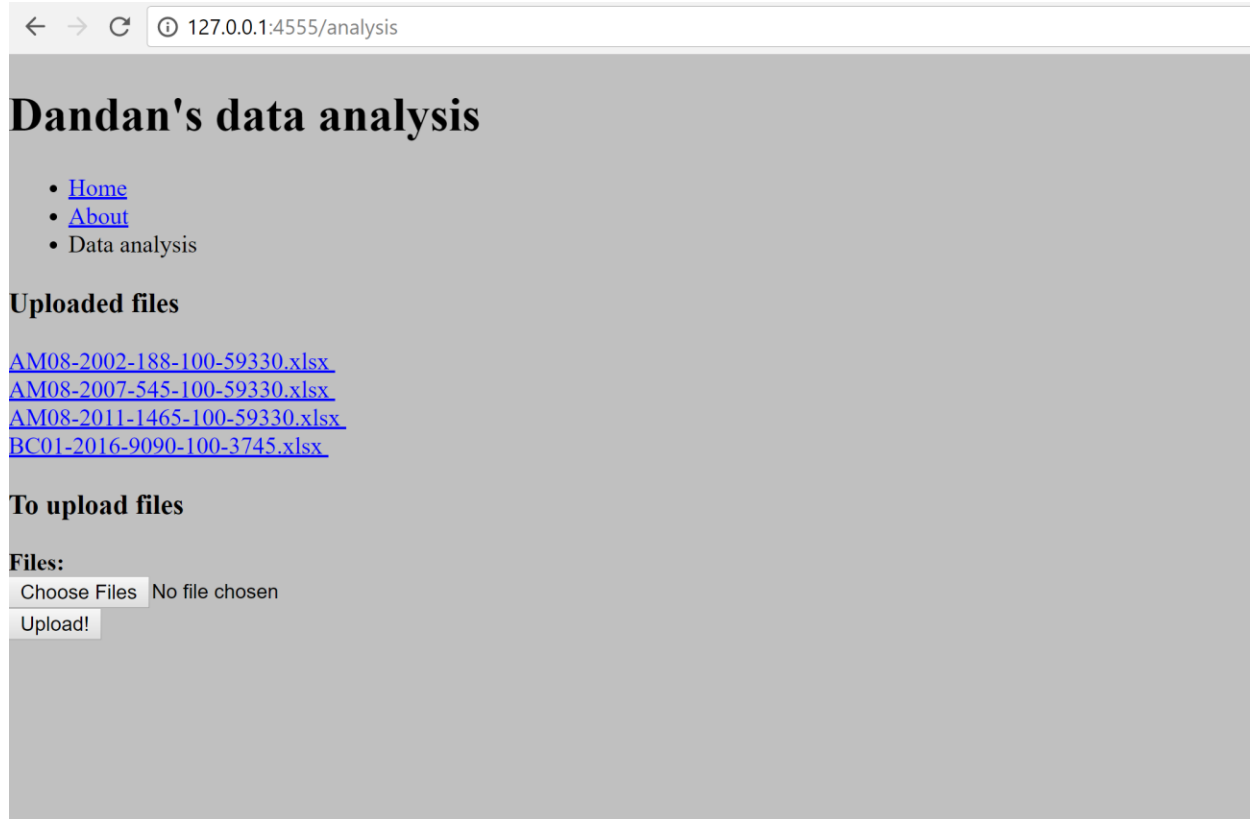
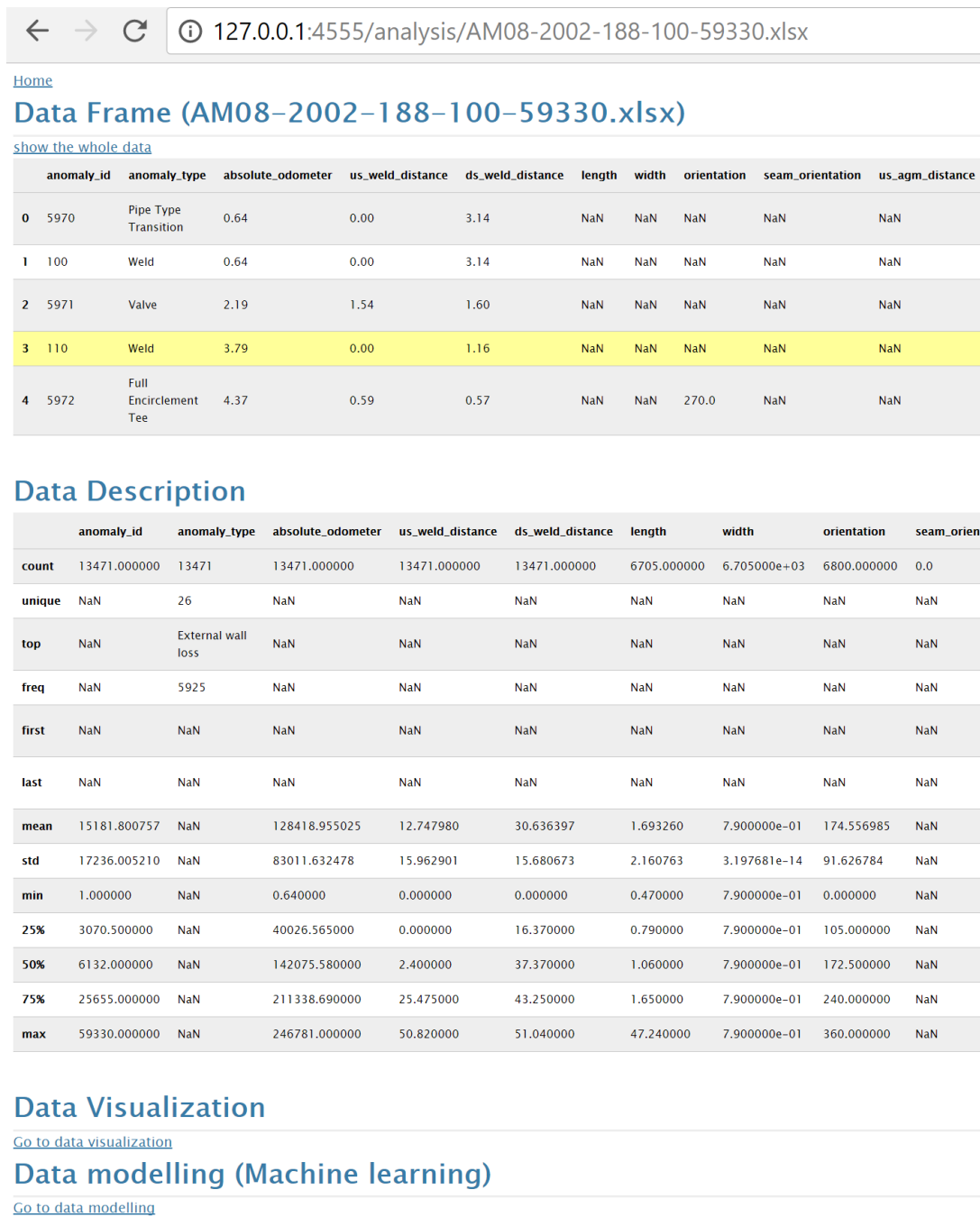


Figure 14 Data loading

After uploading, the user can click on the file name (with a hyperlink). It will show as the following page Figure 15 Statistic summary. It is the first 5 rows with header row. User can also click 'show the whole data' to see the complete data file. The statistic summary of each variable (column) from the data file is also shown on the same page.



In order to see more details of the data, this project has provided various functionalities in Figure 16 Data frame:

1. Side navigation bar to switch between 'data view', 'summary statistics', 'visualization', and 'data modeling'.
2. The ability to select how many entries of the data to show.
3. The ability to sort each column of the data
4. The search box to filter the whole data table
5. Individual search box to filter each column
6. Pagination to see the rest of the data.

File tables x

127.0.0.1:4555/analysis/Auto_data.xlsx/view_whole_data

Home About Data analysis Contact

Data Frame (Auto_data.xlsx)

Show 10 entries Search:

mpg	cylinders	displacement	horsepower
9.0	8	304.0	193
10.0	8	360.0	215
10.0	8	307.0	200
11.0	8	318.0	210
11.0	8	429.0	208
11.0	8	400.0	150
11.0	8	350.0	180
12.0	8	383.0	180
12.0	8	350.0	160
12.0	8	429.0	198

Search mpg Search cylinders Search displacement Search horsepower

Showing 1 to 10 of 397 entries Previous 1 2 3 4 5 ... 40 Next

Contact Me
Email: dandan2@ualberta.ca
Phone: 7809528637

© 2017 Ualberta

Figure 16 Data frame

3.2 Data visualization

A list of the variable names will show up on the visualization page and the user is able to choose which variable(s) to do the visualization. It can automatically detect the data type of each variable (e.g., numerical or categorical).

This project offers Histogram (which is good for numerical variables), Bar chart (good for categorical variables), Box plot (where we can easily see max, min, median,

Q1, and Q3), Time series (which gives us the variable changing trend by time), and scatterplot (where we can get the association between two variables)Figure 17 Graphs. Here are some plots from this project:

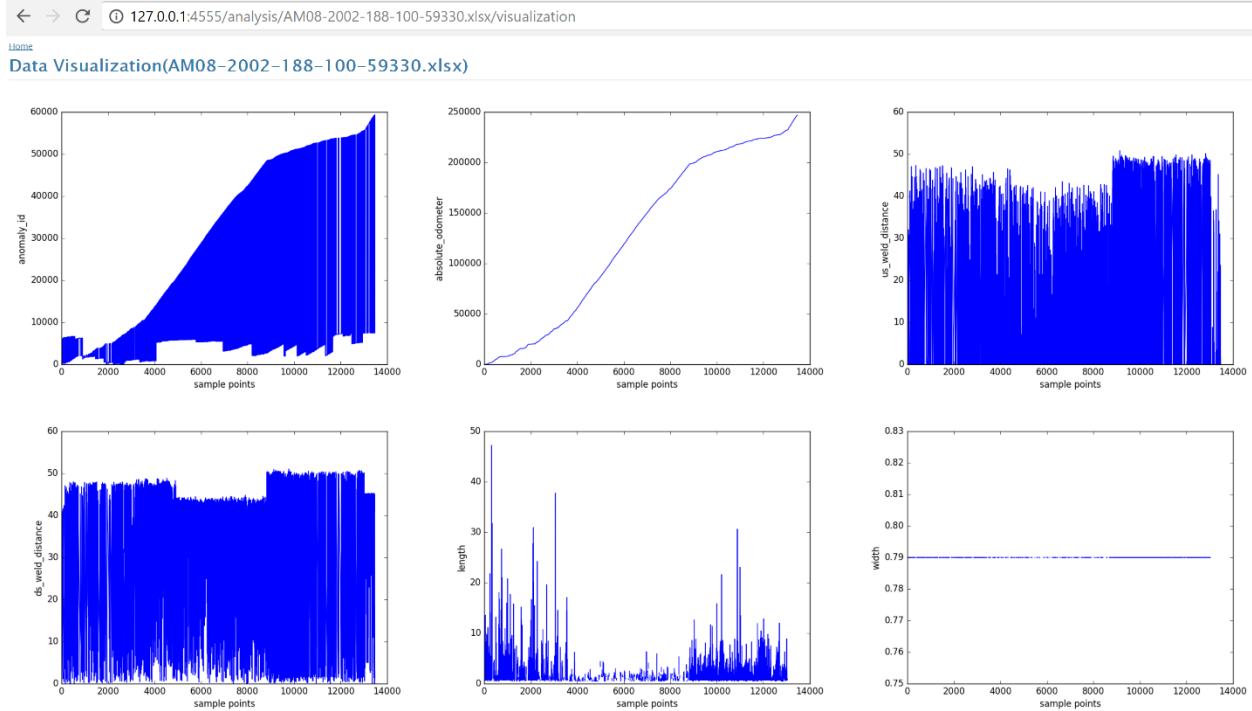


Figure 17 Graphs

3.3 Data modelling

If the user has data (including predictor and response variables) and wants to build a model out of them to predict the response, this project has this functionality to train the model and make predictions.

For machine learning algorithms, it includes two modeling methods: Linear Regression and Random Forest. They are the most common/popular methods. This project uses Python (Flask), so I imported the scikit-learn package. It is easy to use these machine learning algorithms included in this library.

4 Website deployments

Until now, the web application just lived on my computer. We can only access it from my computer, since all the documents related to this application are local. We want to deploy this application to put it online in order to make all the users be able to access it from their own device using a certain URL. Actually, deploying an application is nothing but to find a computer to host the file for us. We can access those files any time from any device. Those computers are also called servers which are managed by different companies like Amazon, Google, or Microsoft. The reliability and cost efficiency are very important when we choose the server. Considering those things, I chose to use Heroku because it is really developer friendly and they also offer a free tier and they also have their own database.

For this project, I put the back end, the database and the front-end (HTML, CSS, and JavaScript) on Heroku but on different servers in order to increase the reliability. This means if one server crashes we still have the other two parts up and running.

There are a few different languages (Figure 18 Different languages for deployment) we can choose when we deploy our application like Ruby, PHP, Node.js, Python, Java, and so on. We need to choose the one we used to develop the application to deploy the application.

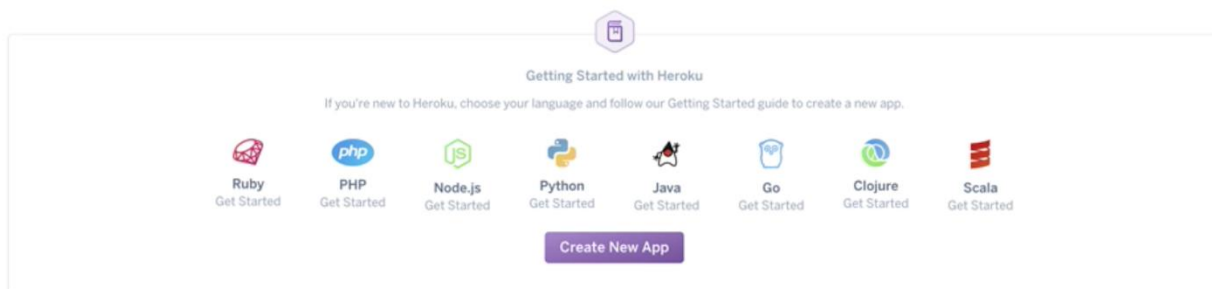


Figure 18 Different languages for deployment

There are also a few ways we can deploy the application. I chose to deploy the application with Heroku Git. Before we deploy, there are two prerequisite things we

need to install which is Git and Heroku CLI. And then we need to add the application to Git, using the follow command. My project name is dandancapstone.

Install the Heroku CLI

Download and install the Heroku CLI.

If you haven't already, log in to your Heroku account and follow the prompts to create a new SSH public key.

```
$ heroku login
```

Create a new Git repository

Initialize a git repository in a new or existing directory

```
$ cd my-project/  
$ git init  
$ heroku git:remote -a dandancapstone
```

Deploy your application

Commit your code to the repository and deploy it to Heroku using Git.

```
$ git add .  
$ git commit -am "make it better"  
$ git push heroku master
```

Existing Git repository

For existing repositories, simply add the `heroku` remote

```
$ heroku git:remote -a dandancapstone
```

If there is something wrong, we can use the command ‘Heroku logs --tail’ to see the log file and figure out what is happening and then correct the error. I had a problem (Figure 19 The problem that I had): the web process failed to bind port within 60 seconds of launch. This is because when I setup the environment variable I set up a certain port (in my case it is 3000) to listen to, but Heroku has its own port. In order to overcome this problem, we need to change our command to

`app.listen (process.env.port || 3000,...)`

which means if we receive a port from the environment that we are running in, then run that port, otherwise run port 3000. Every time we change the command we need to update it so that everything is updated.

```

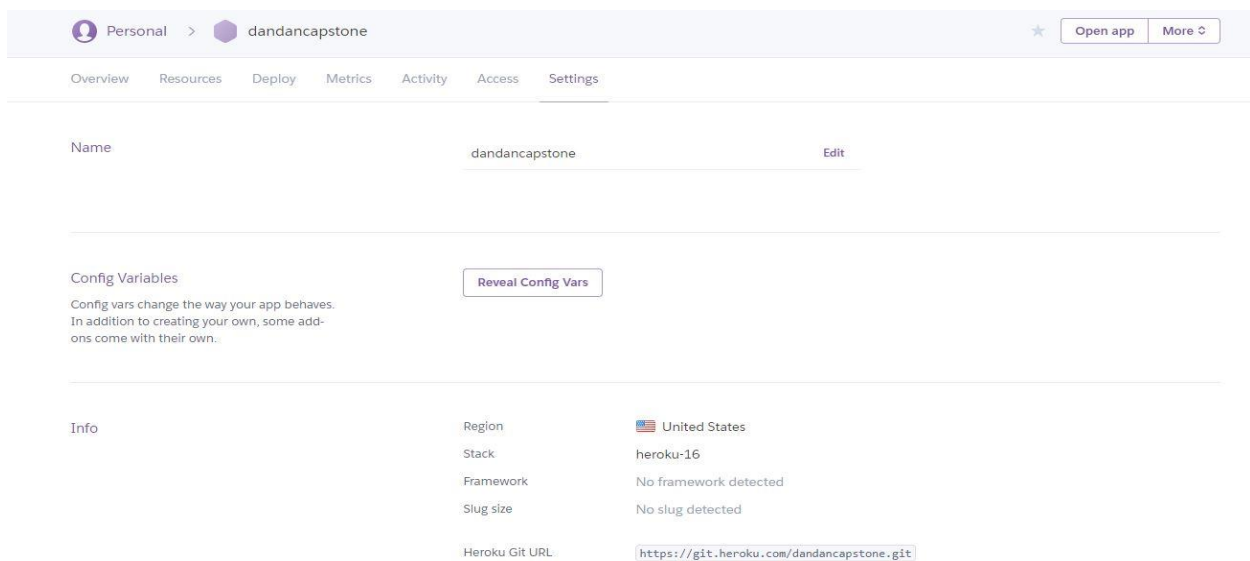
2017-12-21T21:02:19.294336+00:00 app[web.1]: app is running on port 3000
2017-12-21T21:02:17.876977+00:00 app[web.1]: [nodemon] 1.12.7
2017-12-21T21:02:17.878396+00:00 app[web.1]: [nodemon] to restart at any
time, enter `rs`
2017-12-21T21:02:17.878827+00:00 app[web.1]: [nodemon] watching: *.*
2017-12-21T21:02:17.879712+00:00 app[web.1]: [nodemon] starting `node ser
ver.js`
2017-12-21T21:03:13.487120+00:00 heroku[web.1]: Error R10 (Boot timeout)
-> Web process failed to bind to $PORT within 60 seconds of launch

```

Figure 19 The problem that I had

Since I deployed the front end at GitHub page, I need to go to the front end and change all the local host to the actual endpoint. It means we should tell the front end where the back-end server is so that they can communicate with each other.

Now the front end and the server are connected, but we need the database to store the information. In Heroku they have their own database. When we go to the ‘data’ tag, there are different kinds of database and I chose SQL as database like Figure 21 How to choose database and Figure 22 Install database for the project.



Personal > dandancapstone

Overview Resources Deploy Metrics Activity Access Settings

Name: dandancapstone Edit

Config Variables

Config vars change the way your app behaves. In addition to creating your own, some add-ons come with their own.

Reveal Config Vars

Info

Region	United States
Stack	heroku-16
Framework	No framework detected
Slug size	No slug detected
Heroku Git URL	https://git.heroku.com/dandancapstone.git

Figure 20 How to choose database (a)

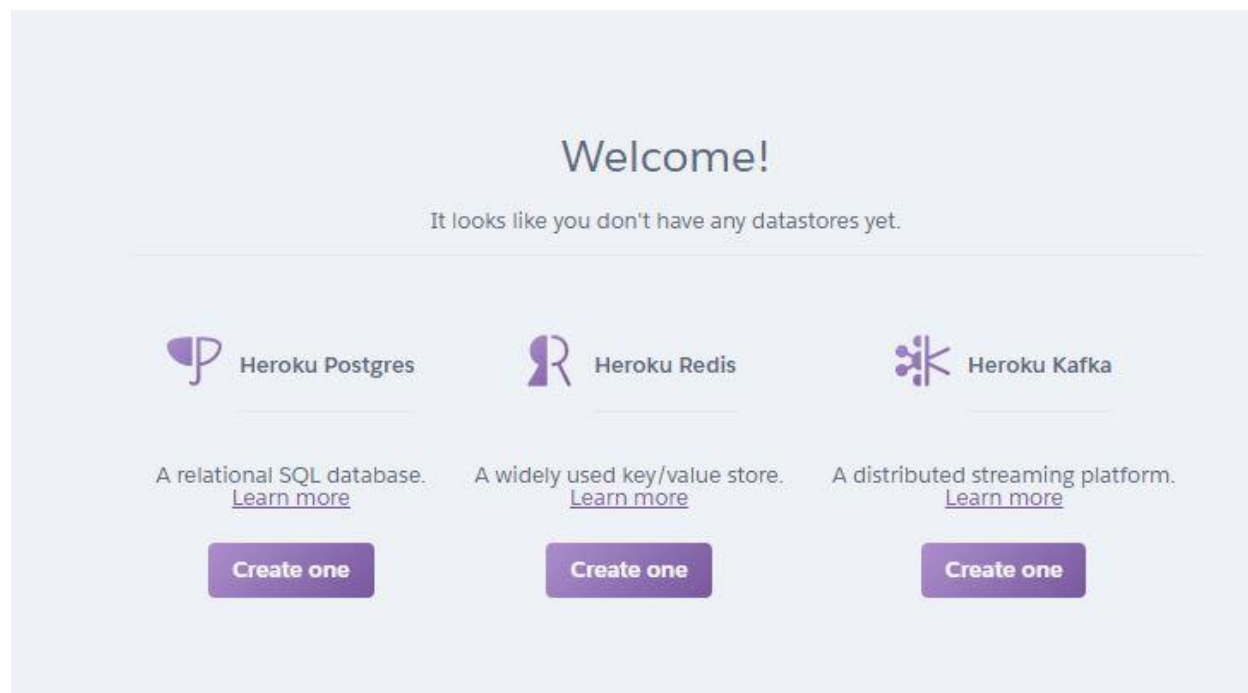


Figure 21 How to choose database (b)

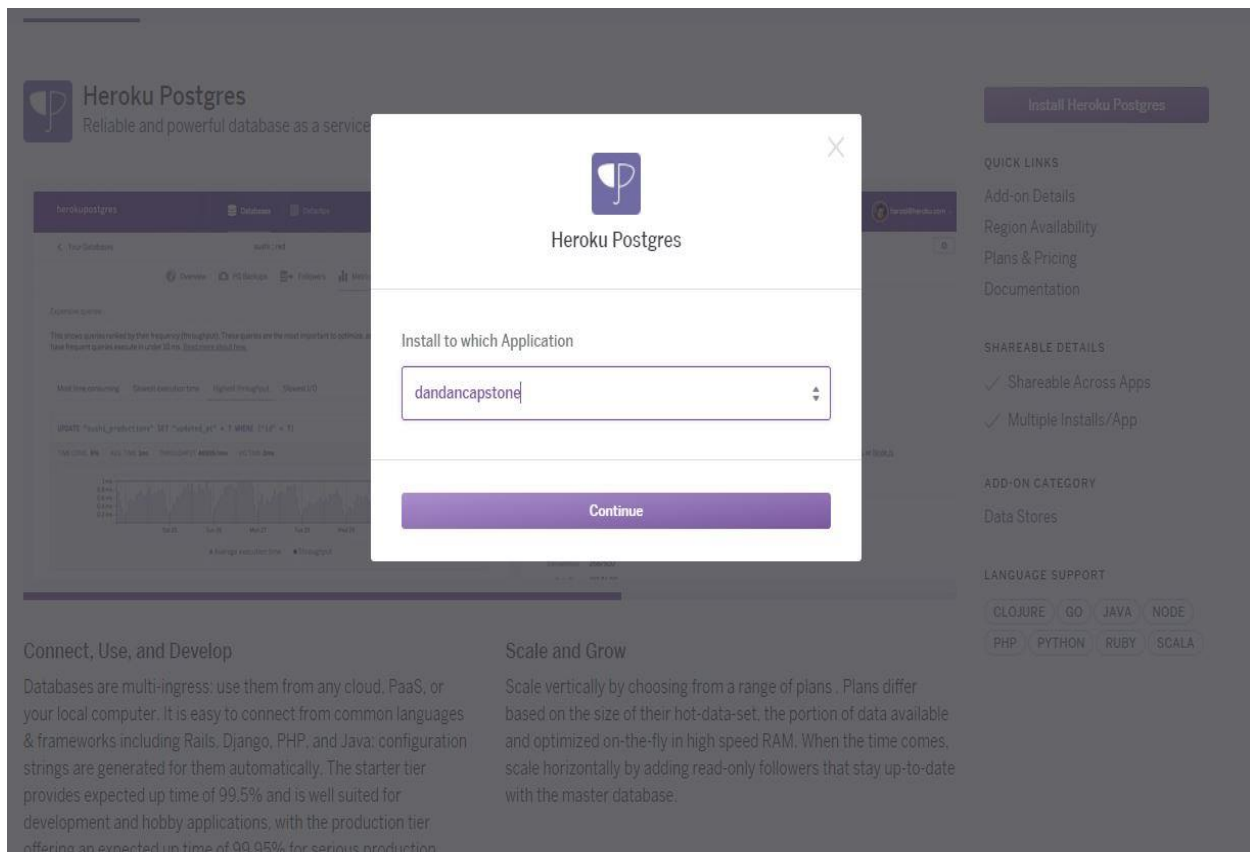


Figure 22 Install database for the project

By following the instruction and run the command below (in Figure 23 Create own database) we can create my own database.

Provisioning Heroku Postgres

Before you provision Heroku Postgres, you should confirm that it isn't *already* provisioned for your app (Heroku automatically provisions Postgres for apps that include certain libraries, such as the `pg` Ruby gem).

Use the `heroku addons` command to determine whether your app already has Heroku Postgres provisioned:

```
$ heroku addons
Add-on                               Plan      Price  State
heroku-postgresql (postgresql-concave-52656)  hobby-dev  free   created
```

If `heroku-postgresql` doesn't appear in your app's list of add-ons, you can provision it with the following CLI command:

```
$ heroku addons:create heroku-postgresql:<PLAN_NAME>
```

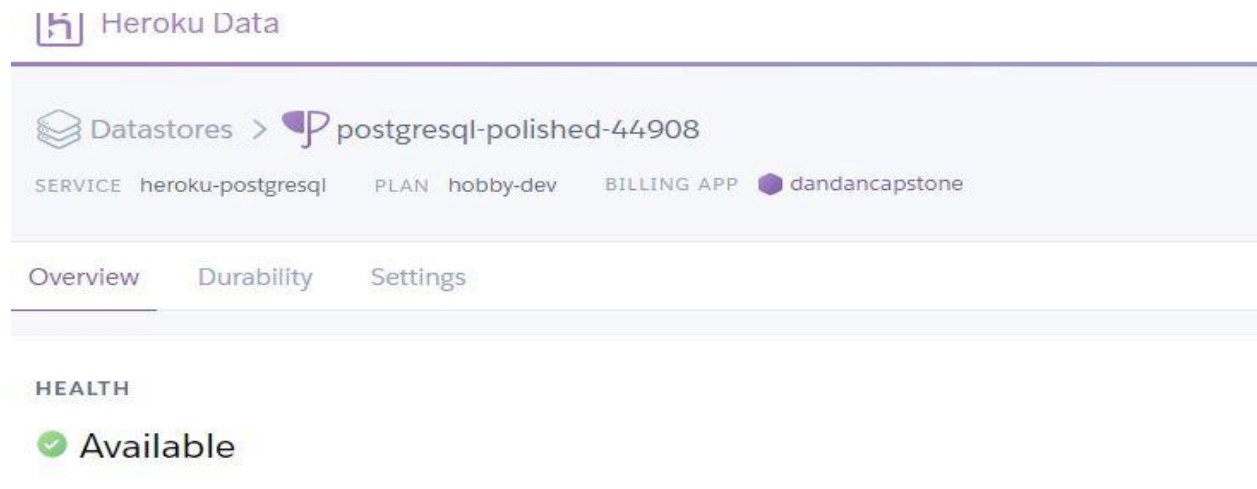
For example, to provision a `hobby-dev` plan database:

```
$ heroku addons:create heroku-postgresql:hobby-dev
Creating heroku-postgresql:hobby-dev on ● sushi... free
Database has been created and is available
! This database is empty. If upgrading, you can transfer
! data from another database with pg:copy
Created postgresql-concave-52656 as DATABASE_URL
```

Depending on the plan you choose, your database can take up to 5 minutes to become available. You can track its status with the `heroku pg:wait` command, which blocks until your database is ready to use.

Figure 23 Create own database

We have the SQL database for the application like Figure 24 Available database.



The screenshot shows the Heroku Data dashboard for a PostgreSQL database. The breadcrumb navigation is "Databases > postgresql-polished-44908". Below this, the service is identified as "heroku-postgresql" with a "hobby-dev" plan and "dandancapstone" as the billing app. The "Overview" tab is selected, showing a "HEALTH" section with a green checkmark and the status "Available".

Service	Plan	Billing App
heroku-postgresql	hobby-dev	dandancapstone

Overview | Durability | Settings

HEALTH

✓ Available

Figure 24 Available database

Now I have database but the server still points to the local database so I need to give the new database location URL to the server to make sure they are all connected together.

By now all three parts are deployed and they know where each other is and they can reach and communicate with each other.

5 Concluding remarks

I learned a lot through the whole process of this project. Not only knowledge but I also found a lot of useful websites and resources that I can use in the future. Moreover, I developed the ability to do the self-learning which I will benefit from in the future as well.

At the beginning of the project, I knew almost nothing about web development. But I have passion in this field, so I chose this one as my capstone project. I chose sublime text as my text editor and set up the environment based on what I like. I started with HTML tags and played with them and then learned CSS properties and applied them on my project. After that I learned JavaScript to add actions on my project. In this stage, I had built the website from scratch because I wanted to see how those things work together. It took me the whole afternoon to figure out how to add landing page and register box and make them work properly. Although I know there are lots of templates about this and all the templates are beautiful, I just wanted to figure out how everything works.

As a web developer, nobody builds things from scratch. And nobody is going to memorize all the HTML tags and CSS properties. What they need is to know how things work together and to understand the code. The most important thing is they know how to build the website to meet the customer need and based on another people's code. NPM (standing for node package manager) was created for developers to be able to share JavaScript code, which means any developer around the world can grab somebody else's code that may help to solve their own problem.

This is a great knowledge base of scripts that people update or improve and build on top of.

I really like this field and I plan to start applying for jobs in this field because I really enjoy working on it. Every time I made a little progress on my web application I felt so happy and proud of myself.

Thanks Dr. Mike for giving me the chance to do this kind of project.

Reference

https://en.wikipedia.org/wiki/Main_Page

<https://www.w3schools.com/>