Summer Poissonnier

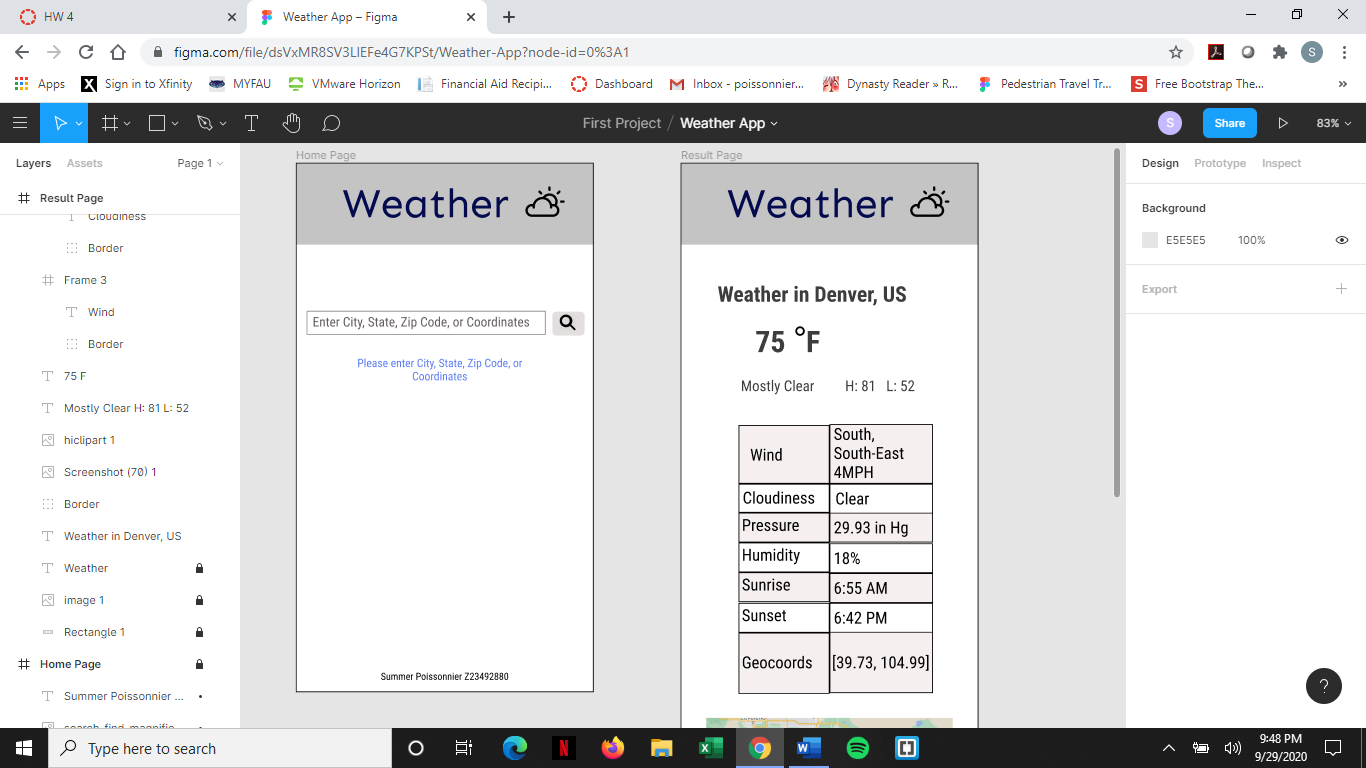
September 29, 2020

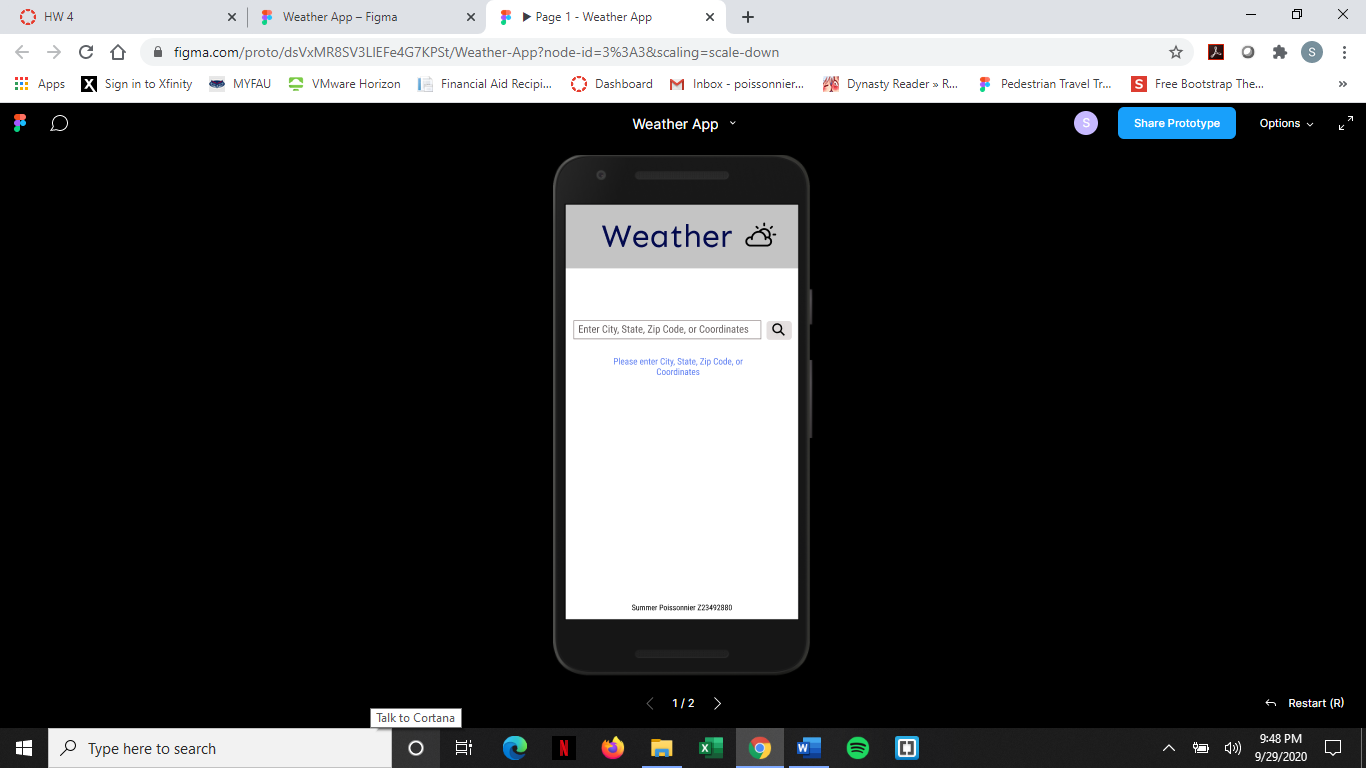
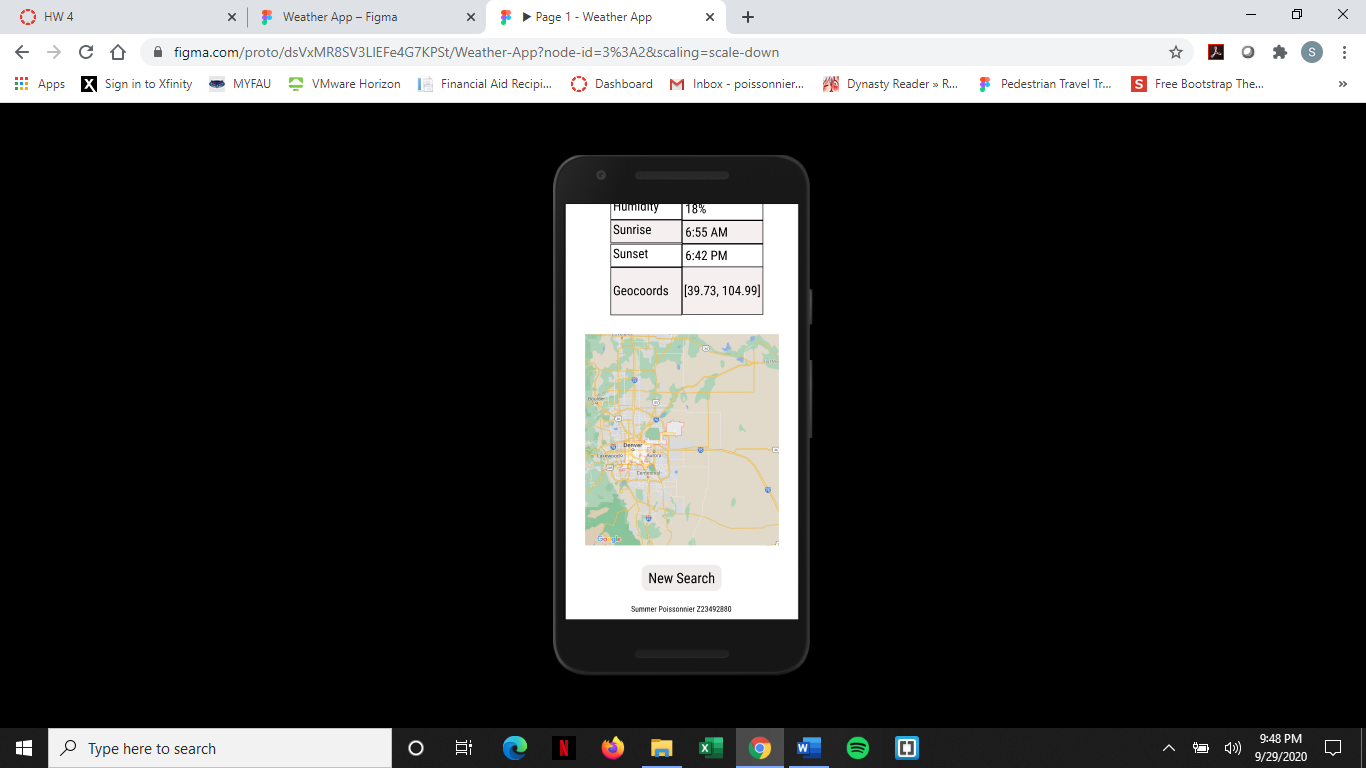
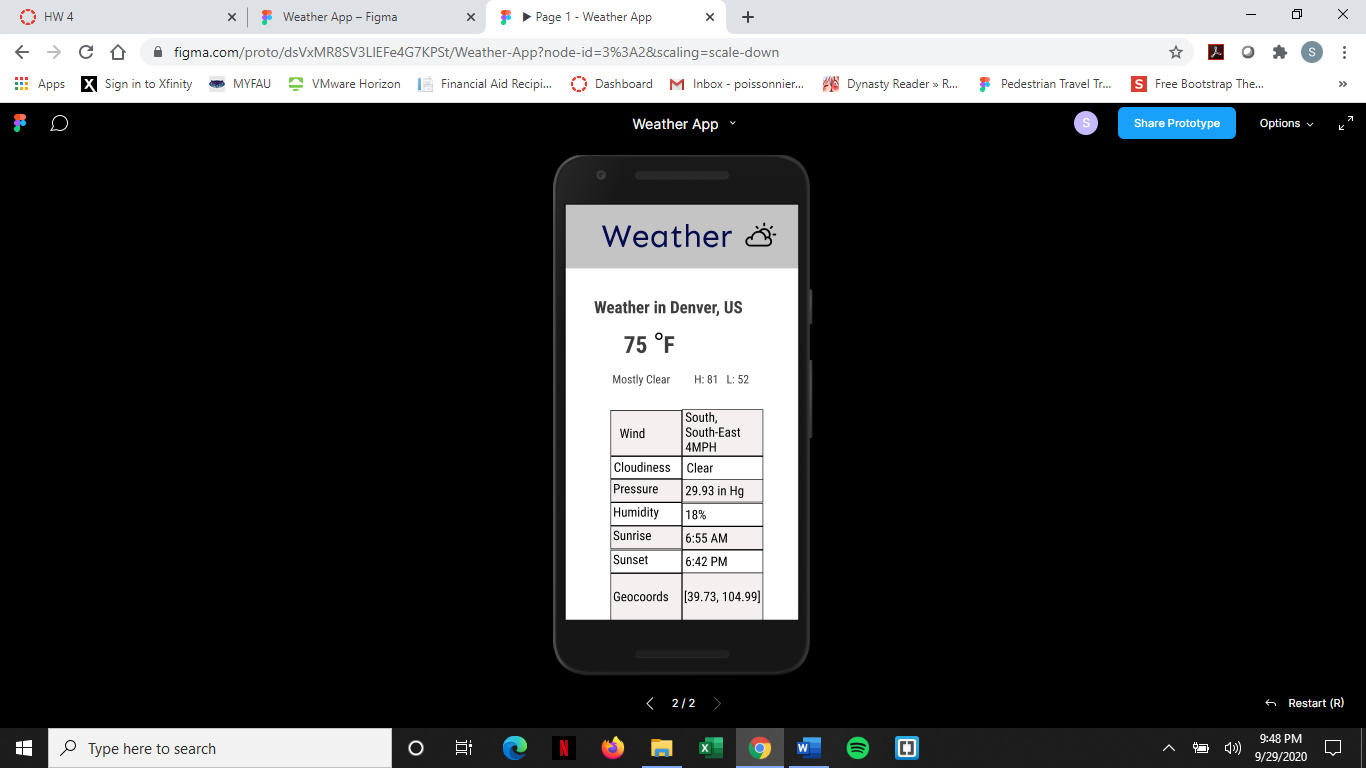
Mobile App Projects

Homework 4

**UI Design**

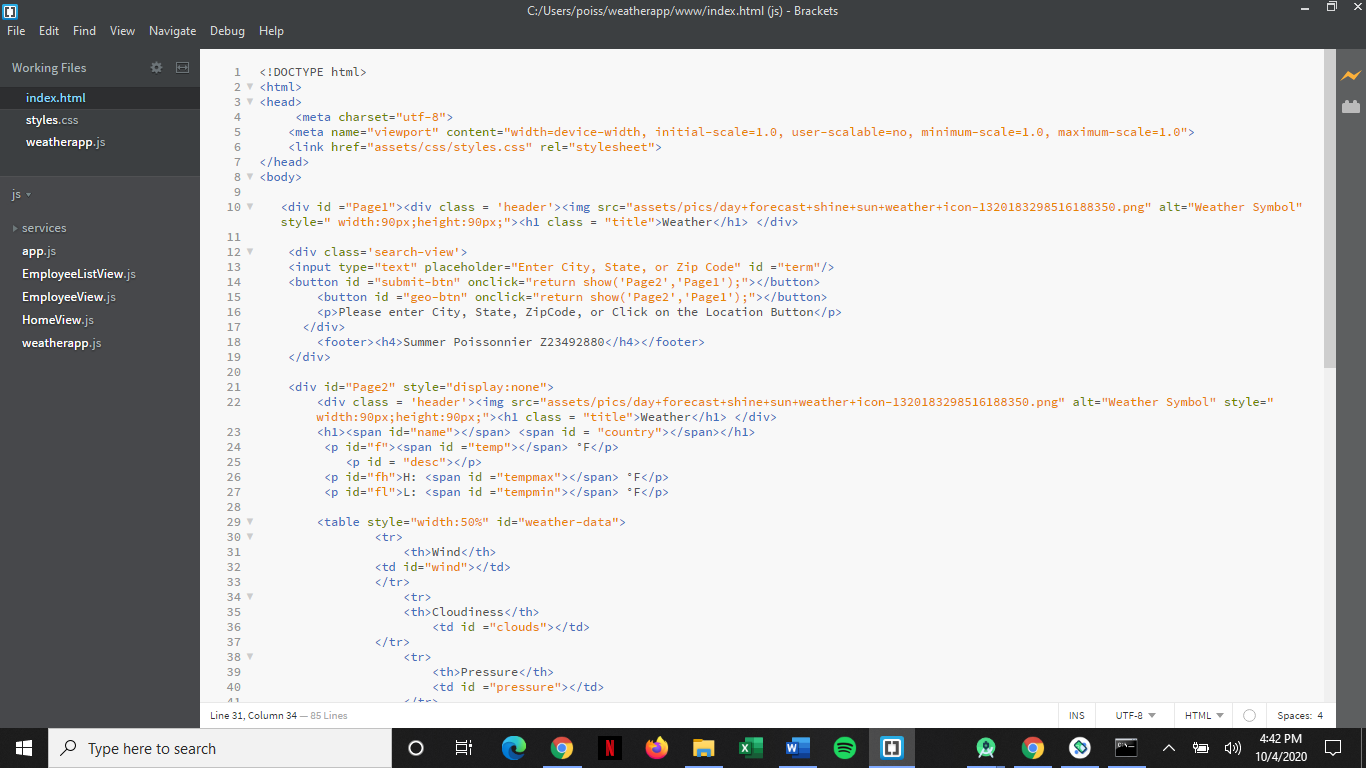
For the UI design of my Weather App, I decided to use Figma. I am familiar with Figma because I have used it with my undergraduate research team. It took me around 2 hours to design my app in Figma to make it the way I wanted.

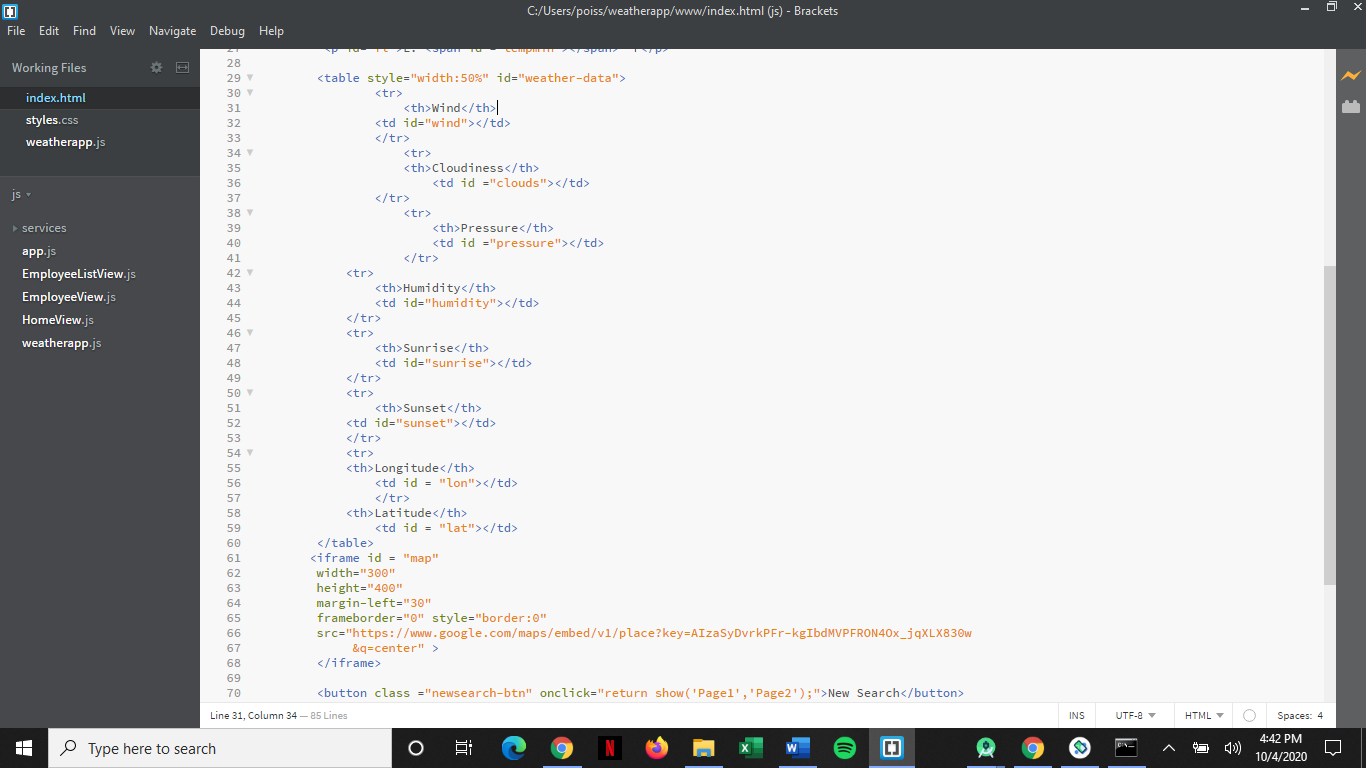




**Index.html Code Explanation**

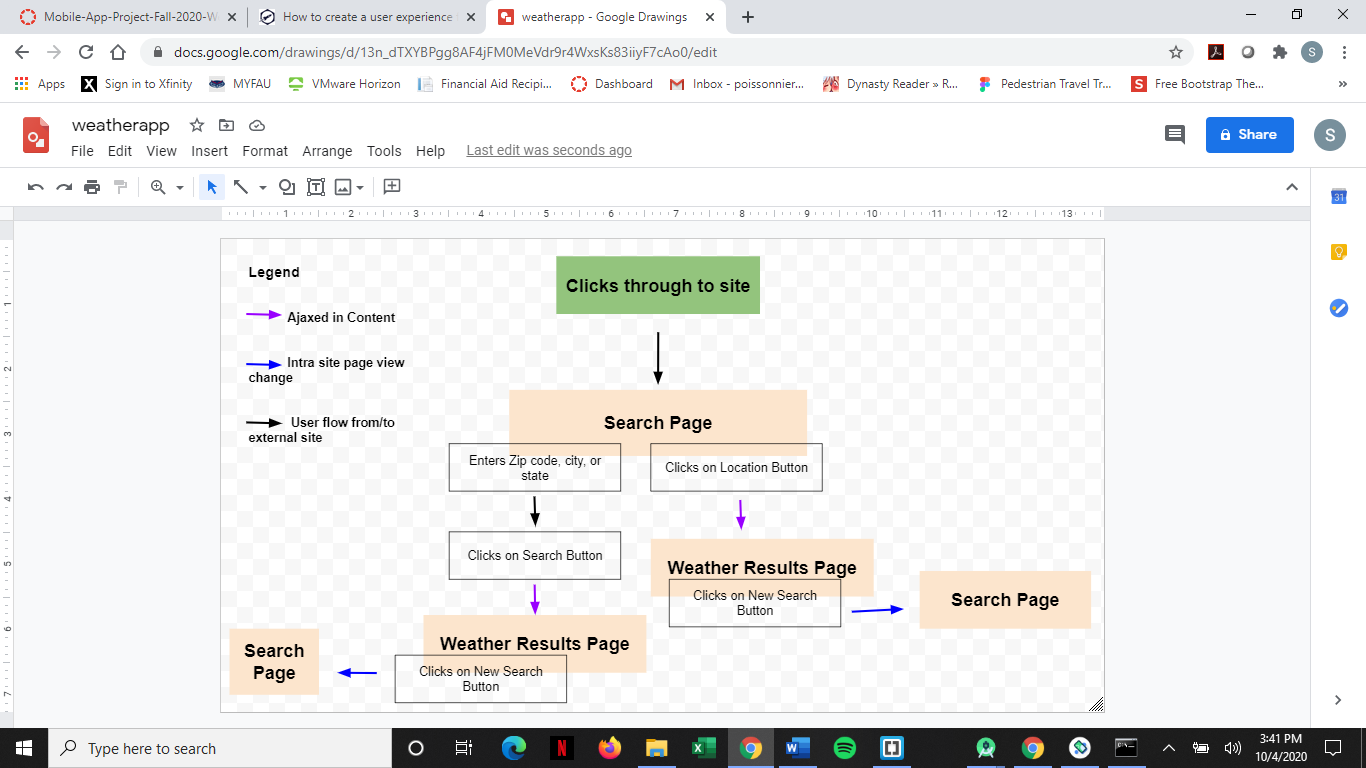
I used an index.html file for the design part of my app. I used a placeholder to make the part of the app where the user would type in the state, zip code, or city. I then created a button that I called “submit-btn” for the user to click on which would then take them to the results page that displays all of the weather information. I designed the submit-btn with a magnifying glass on it. If the user wants to search for the weather via GPS, I created a search button with a search symbol on it and I called the button “geo-btn”. Below the search bar, I also included some text explaining to the user what to do. At the bottom of the search page, I included a footer with my name and Z-number. For the results page, which contained the weather information, I created a <p> tag and <span> tag for the name display and the temperature display. I created “id” names for them so I could link the id names with my weatherapp.js file which would get the information from the weather API. For the 10 weather elements, I created a table in my index.html page and created id’s for the elements and linked them in my js file to get the data from the API. I wasn’t quite sure how to create two different app screens, so I googled it. When I googled it, I found something that helped me create two “pages” in one index.html page.





**UI Flowchart**

I was a little confused on how to go about making this flowchart. I used google docs drawing and created my flowchart for my app in there. I tried to follow the link that was in the PowerPoint for creating flowcharts. I hope my flowchart makes sense and is correct.

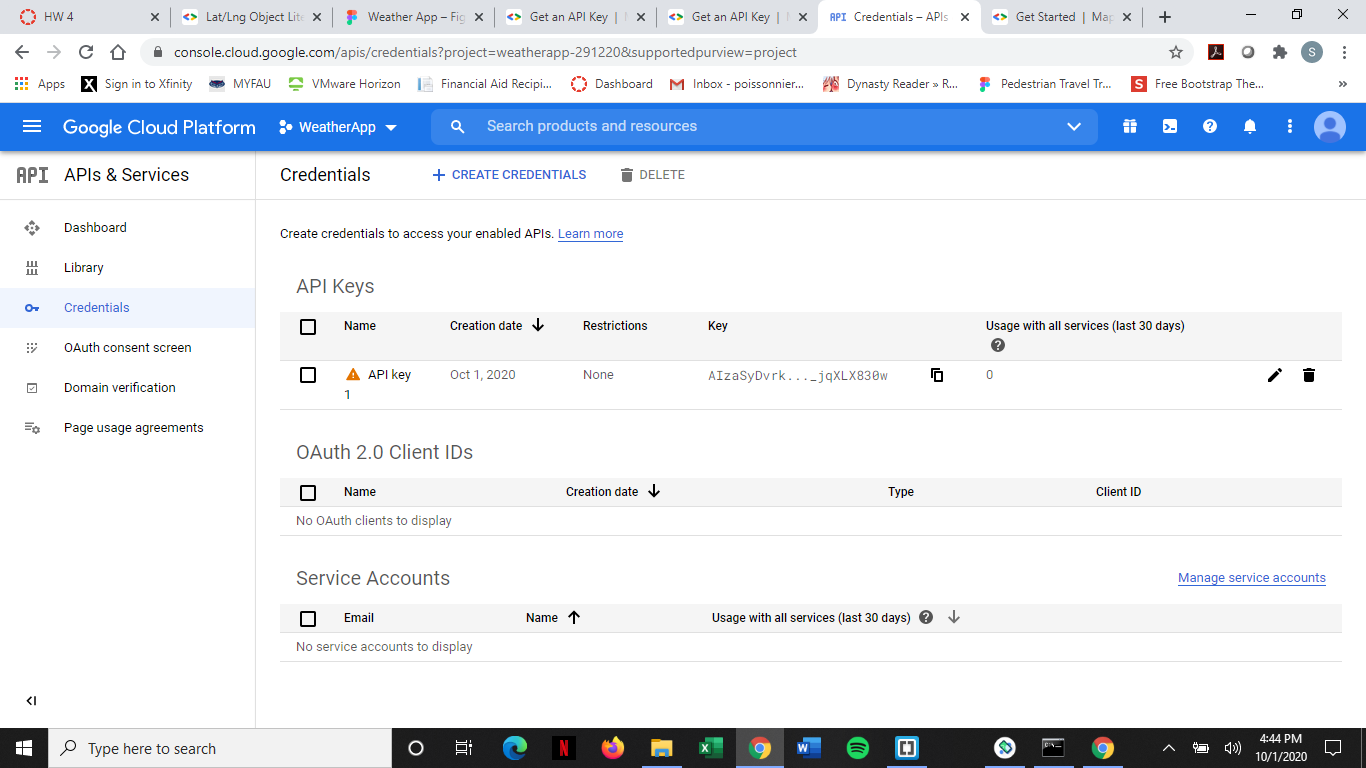
****

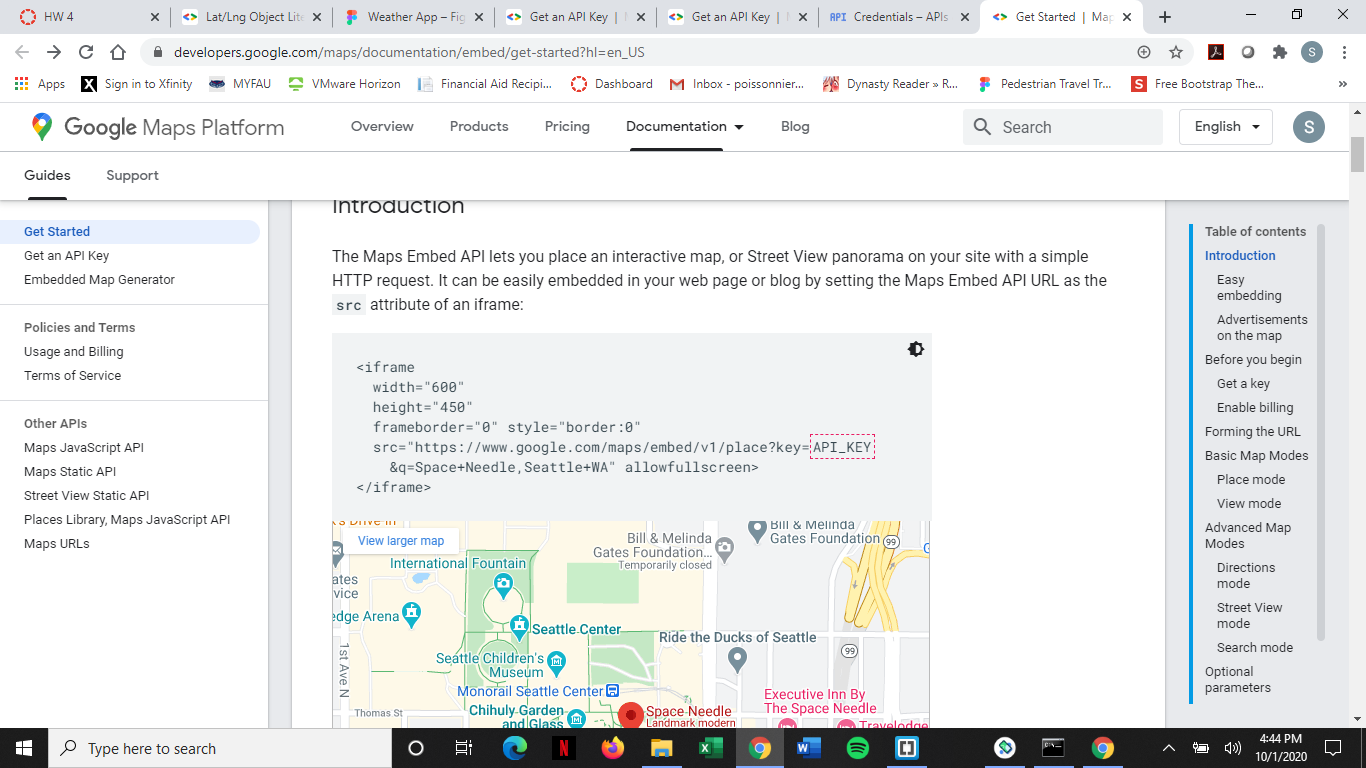
**Open Weather API**

I used the open weather API like I did in HW3 and supported searching for ZipCode, GPS, and City, State.

**Google Maps API**

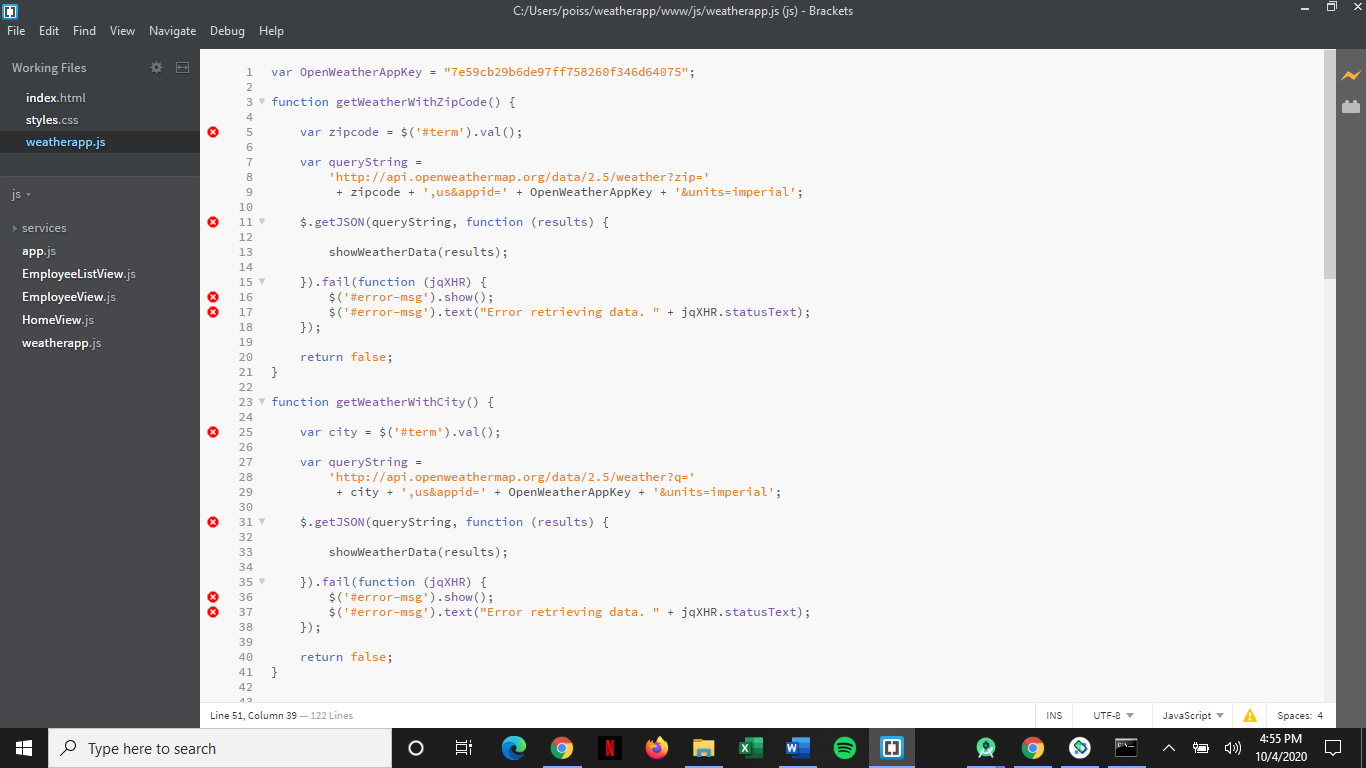
I ran out of time for this because this assignment took me a really long time to do and for some reason, I just could not get this part of the assignment to work properly. I used an embed maps API from google for my map but I did not know how to make it show me my location based on my zip, city, or GPS. To register for the key, I had to make a google cloud account and go to credentials.

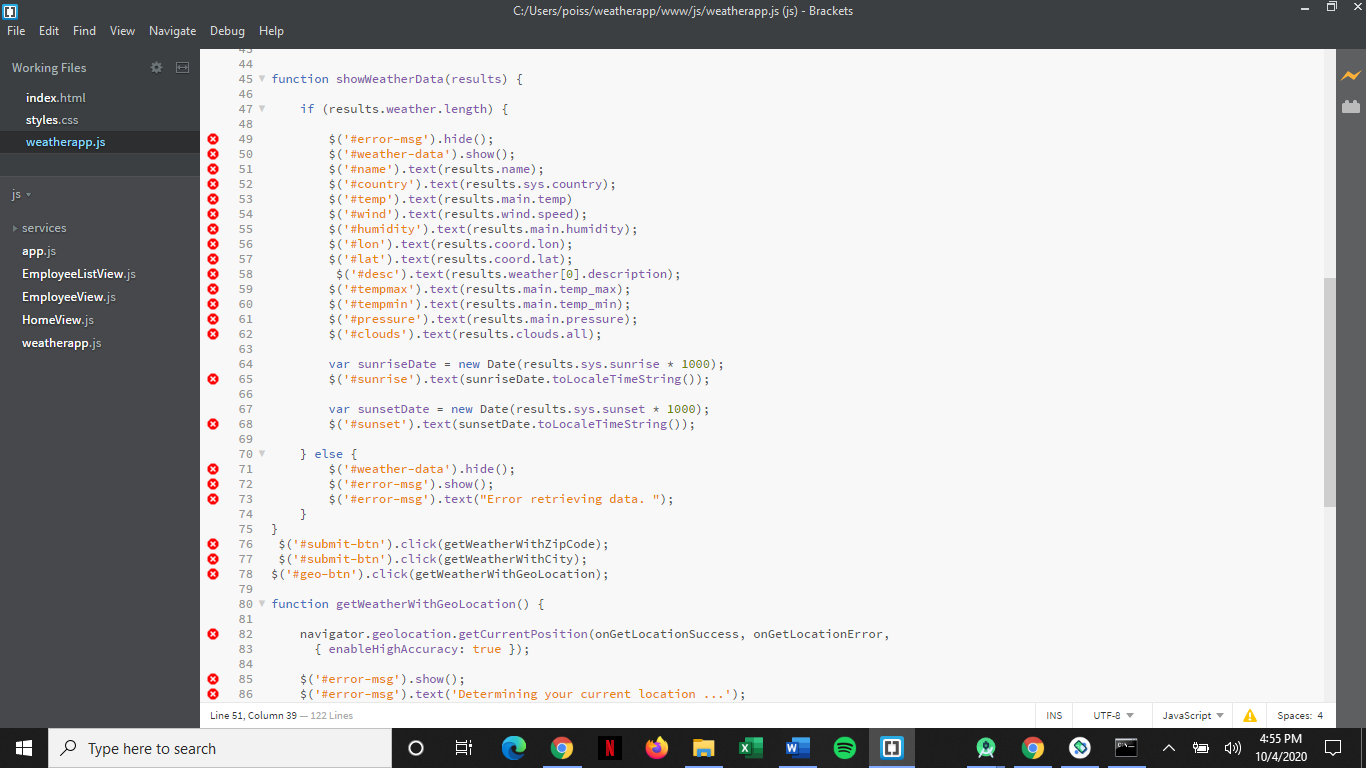


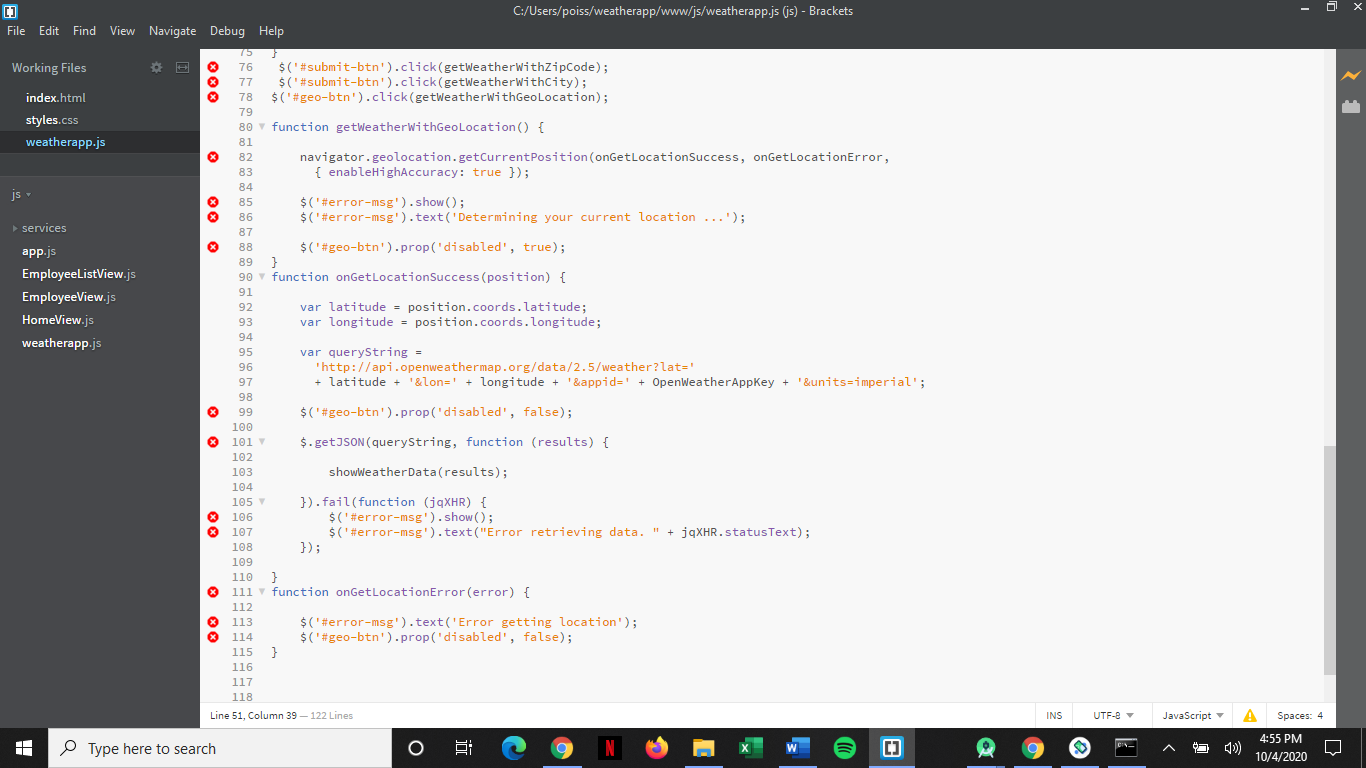


**Weatherapp.js Code Explanation**

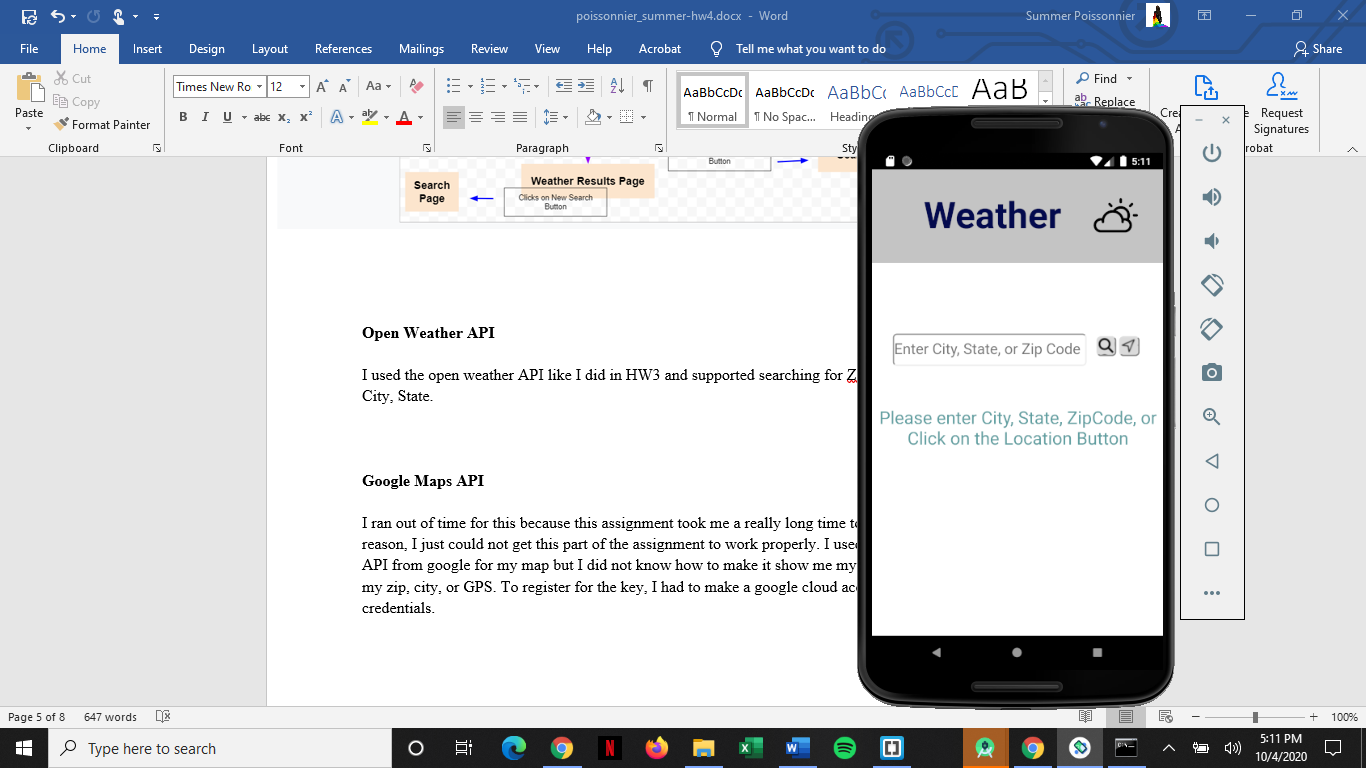
For the js part of the code, I was very confused and I was not sure where to start. I watched a couple YouTube videos and tried to follow what they did. It took me 2 days to figure out how to set my code up and to get everything working since I am not very familiar with js. I also spent a lot of time looking at past assignments and trying to understand what was going on in the code. In my code, I created functions for getting the weather with zip code, GPS, and city/state. Inside the functions, I used the API for the weather. I then created a function I called showWeatherData to display the weather data on my app. I called the id names for the elements that I created in my index.html page to get the information from the JSON data in the weather API. I then tied my buttons (the search and location buttons) to my functions when clicking on them.

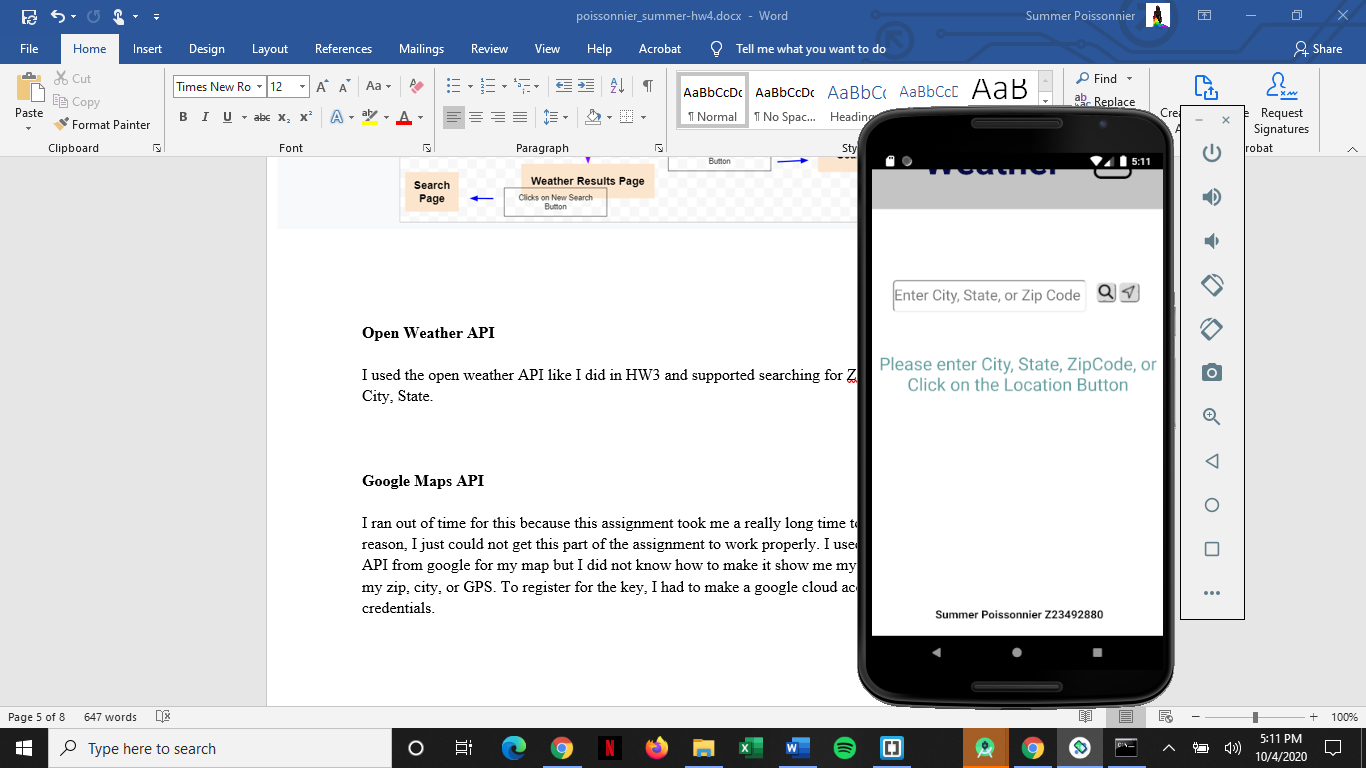
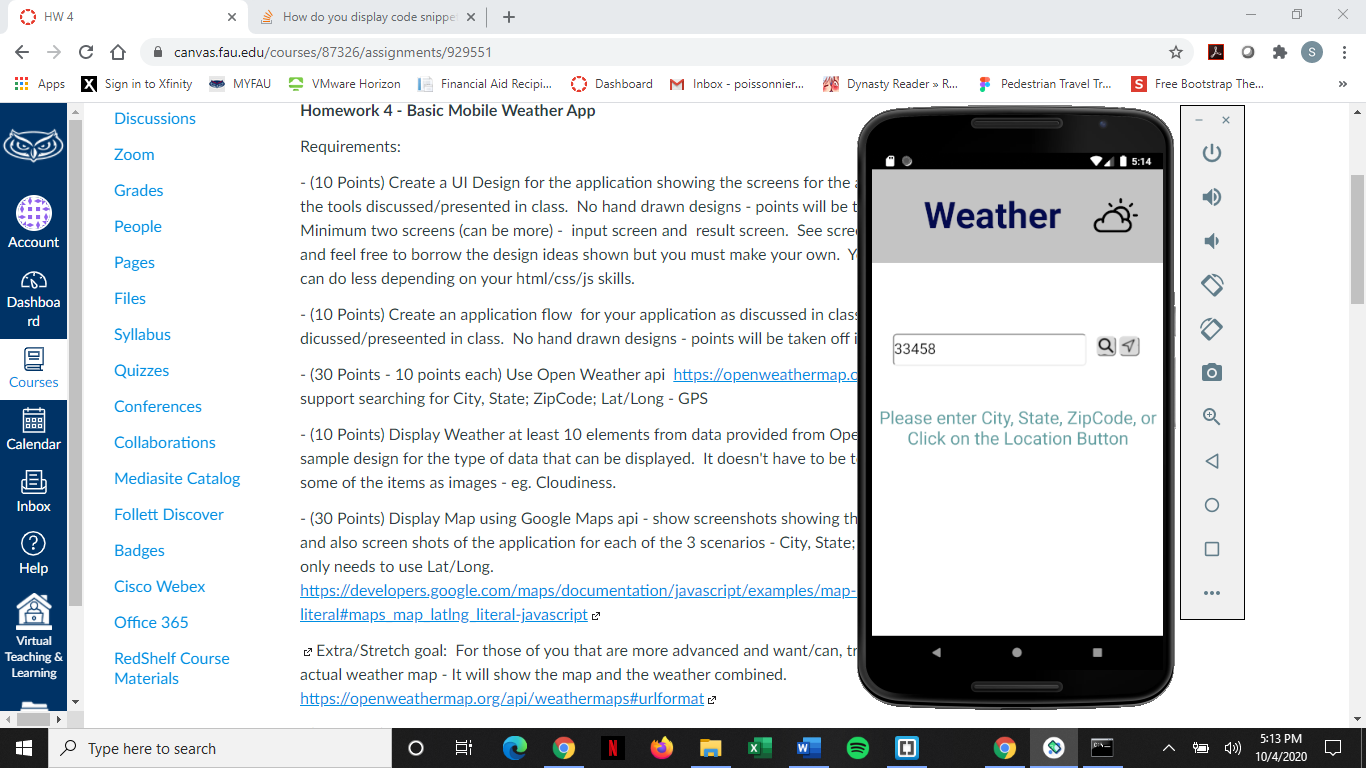


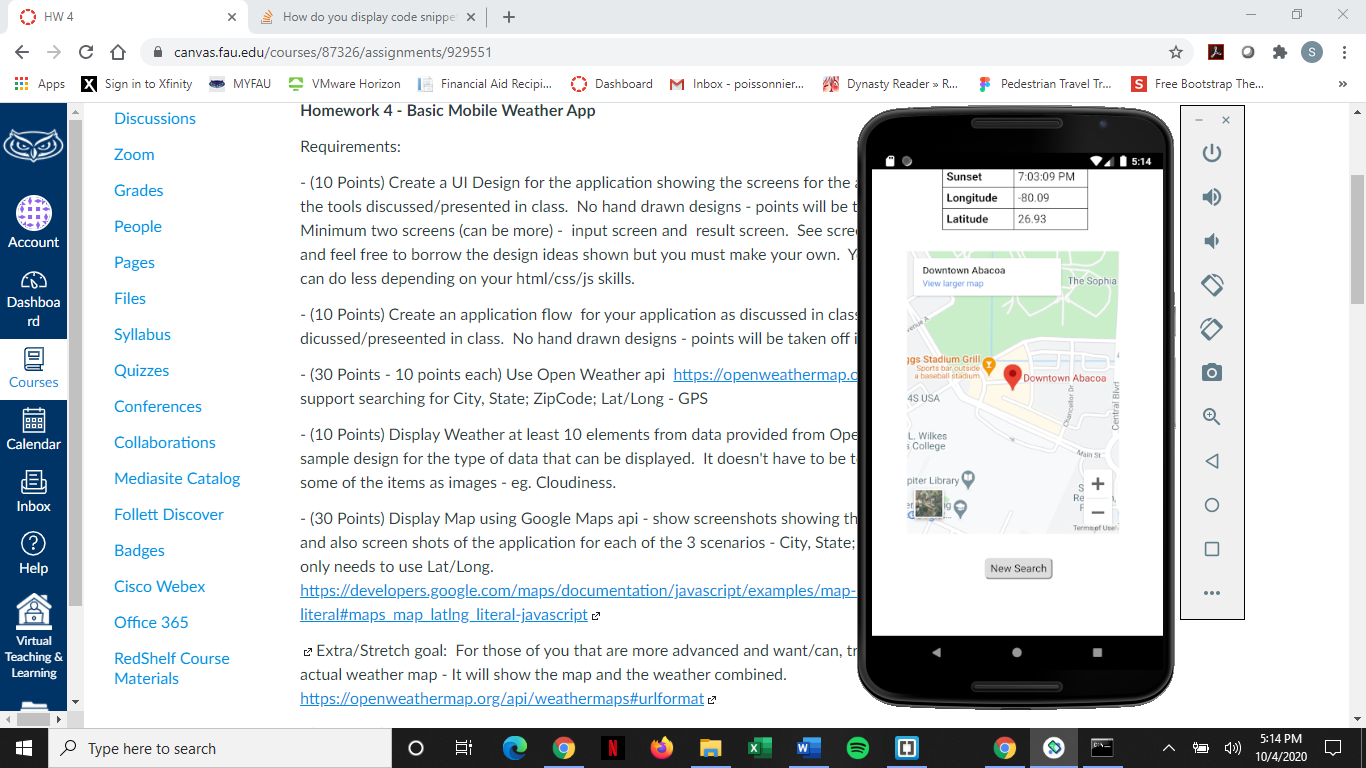
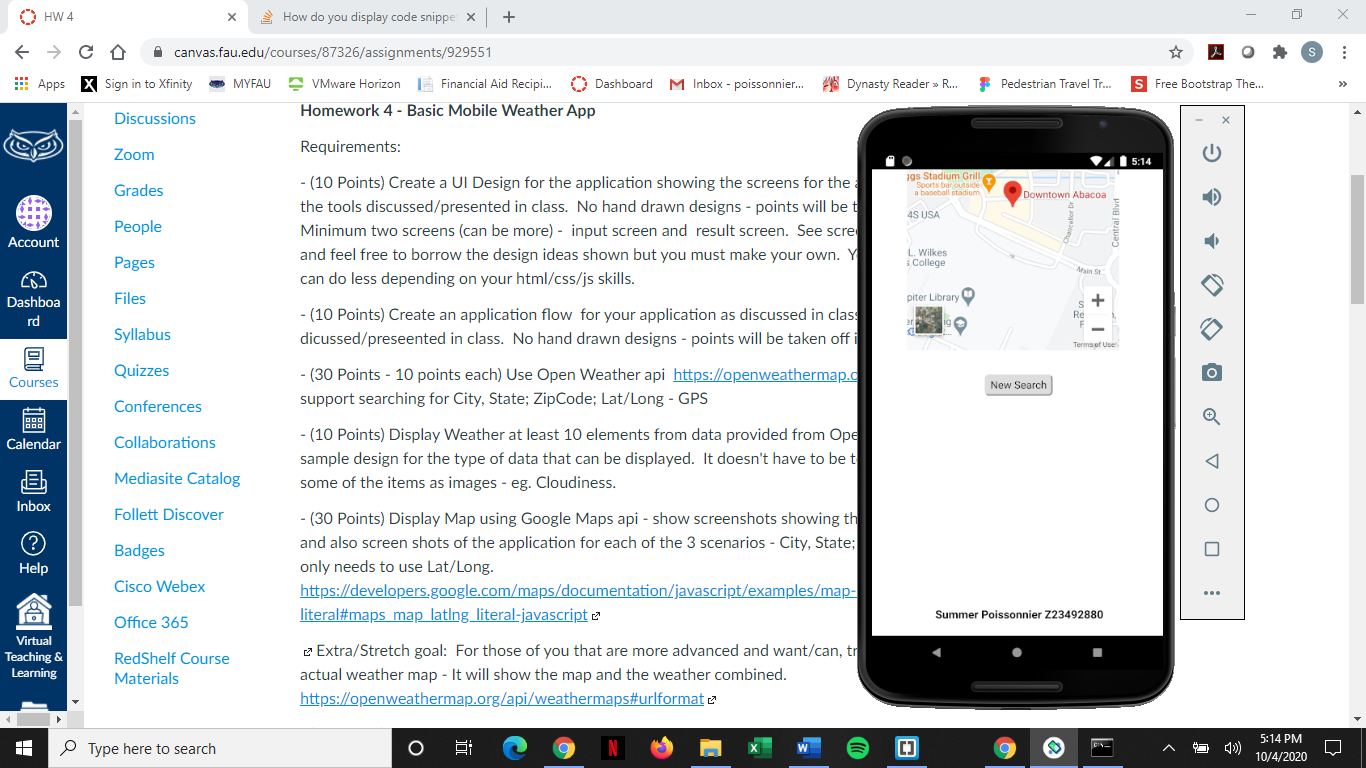
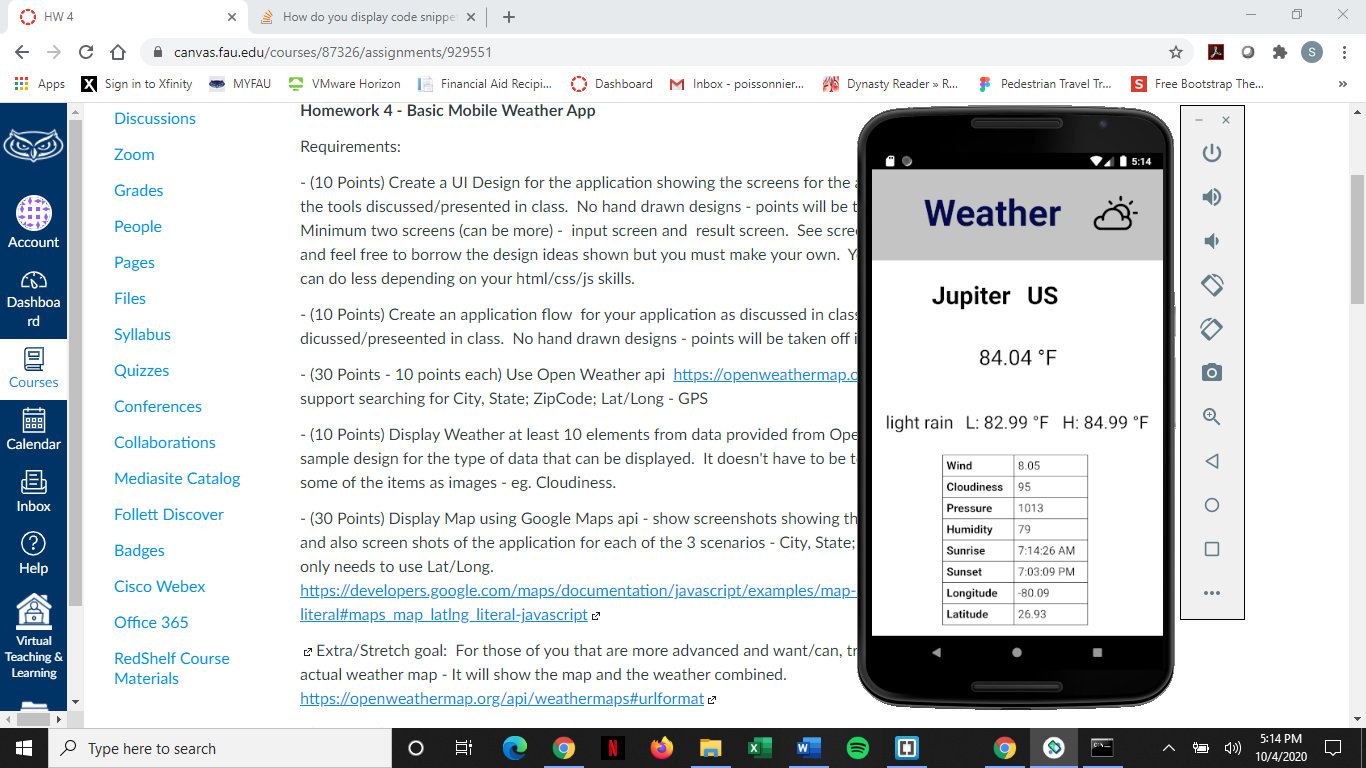
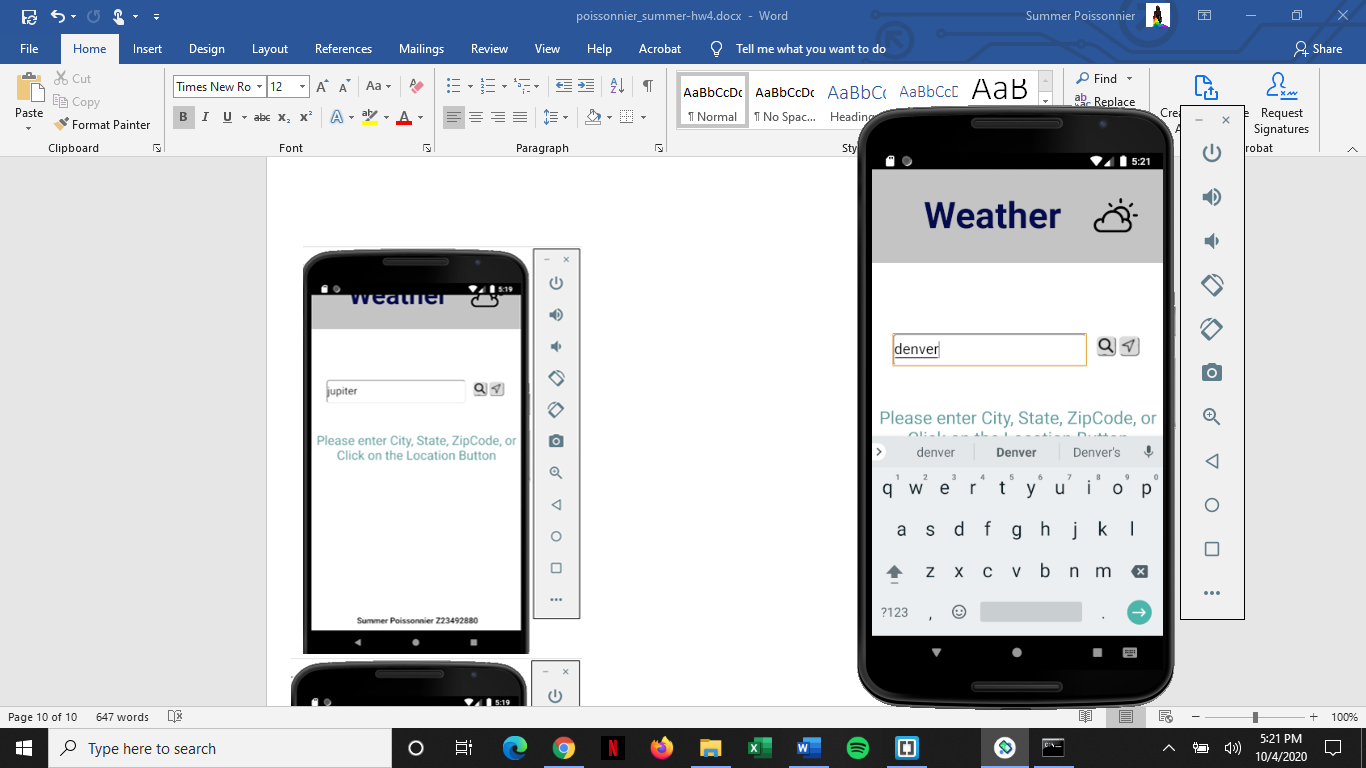
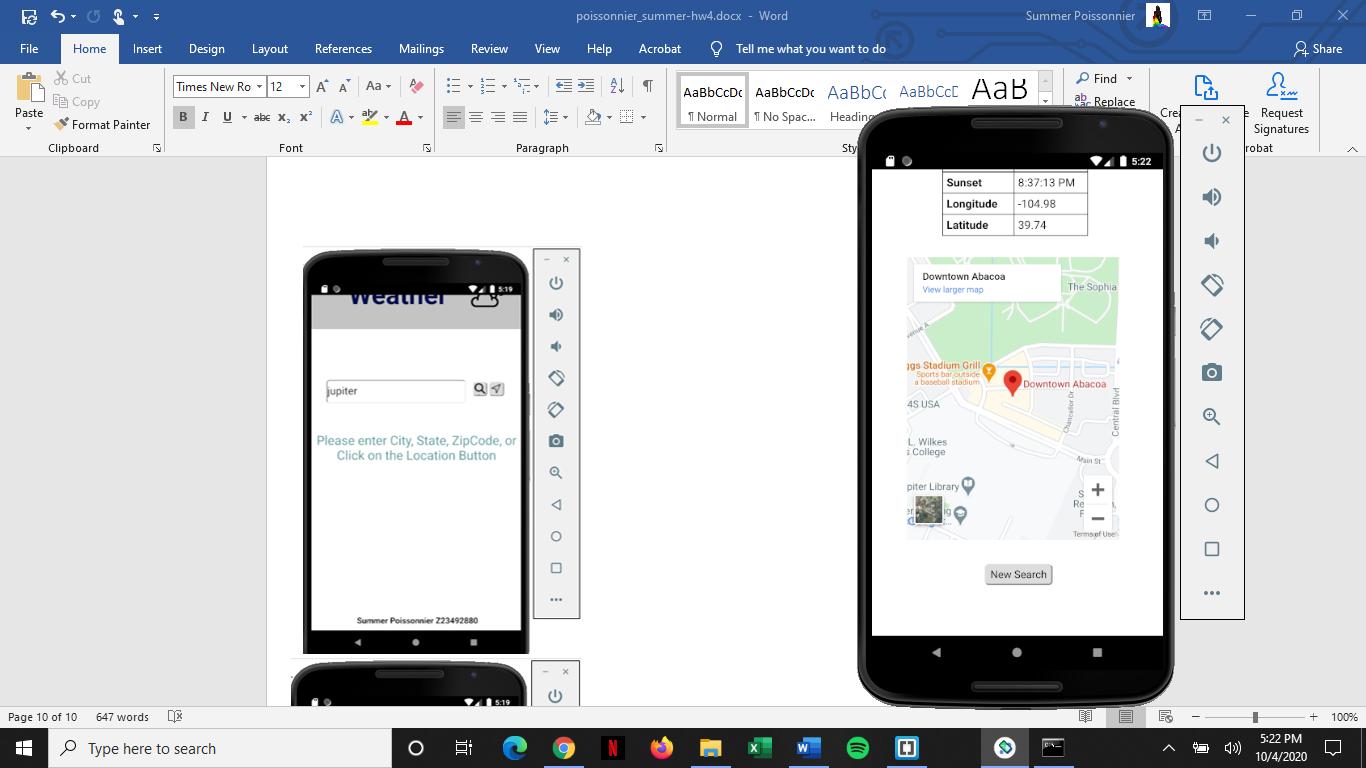


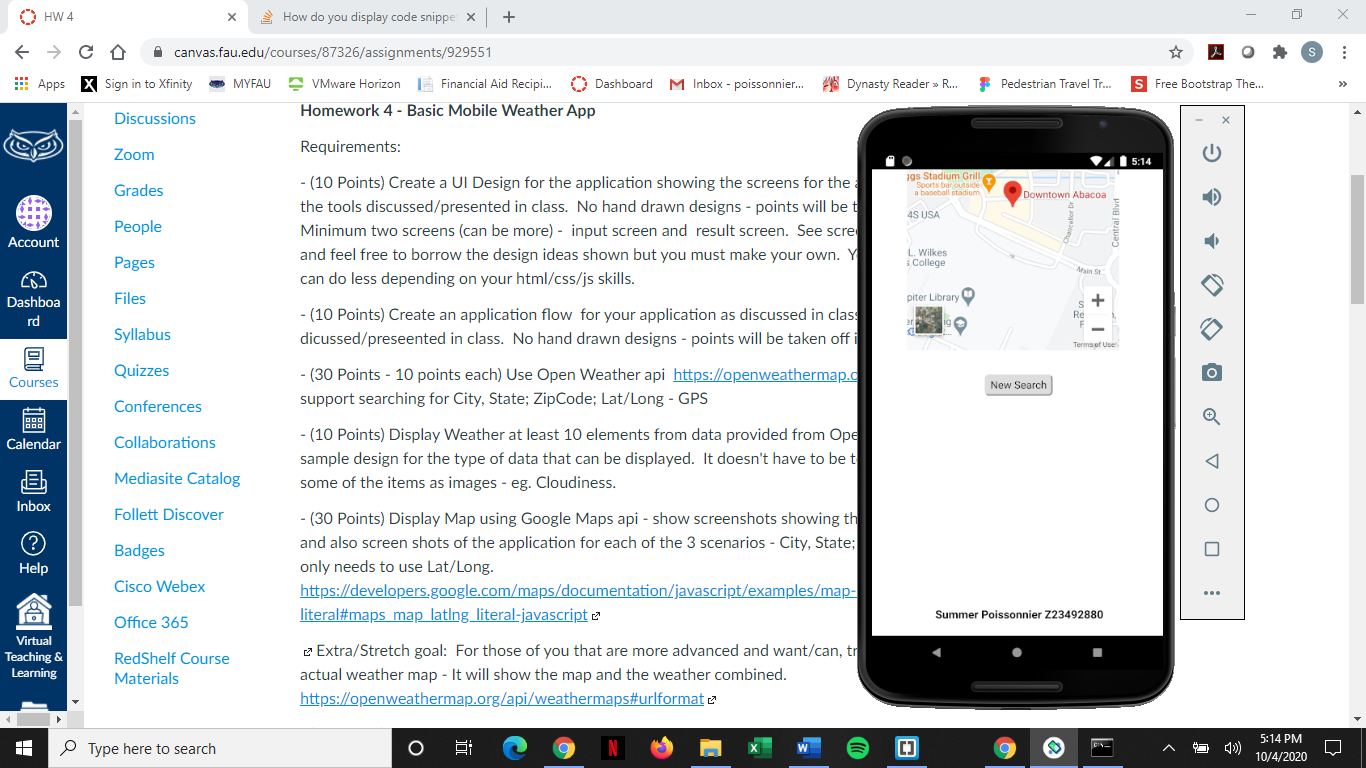
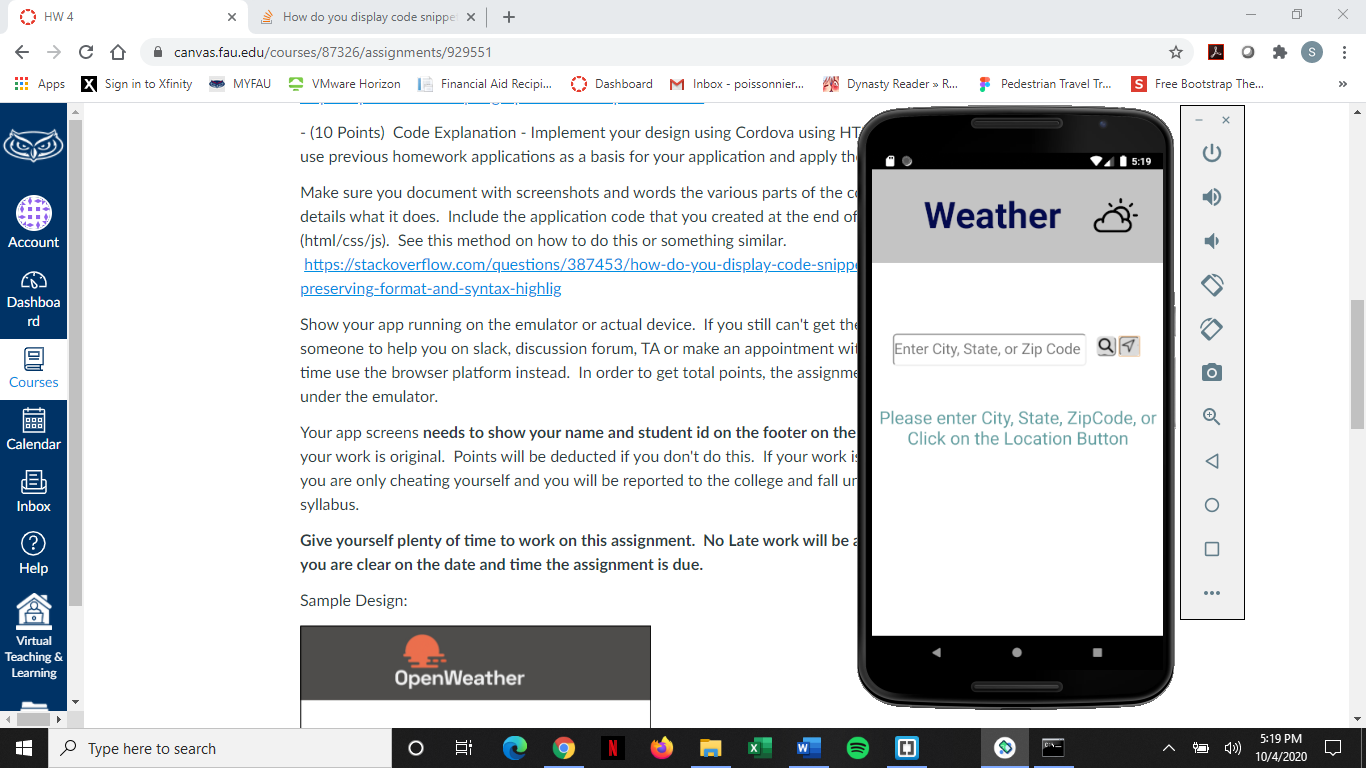


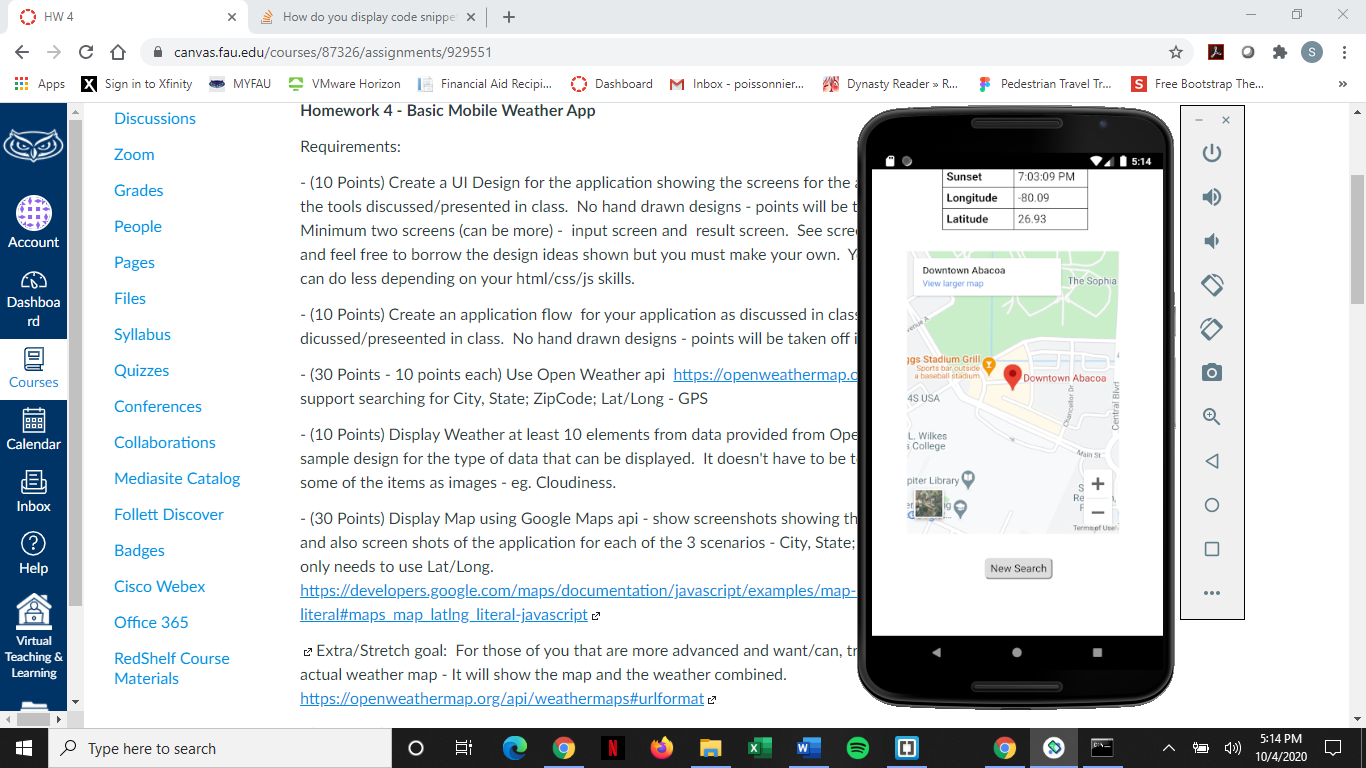
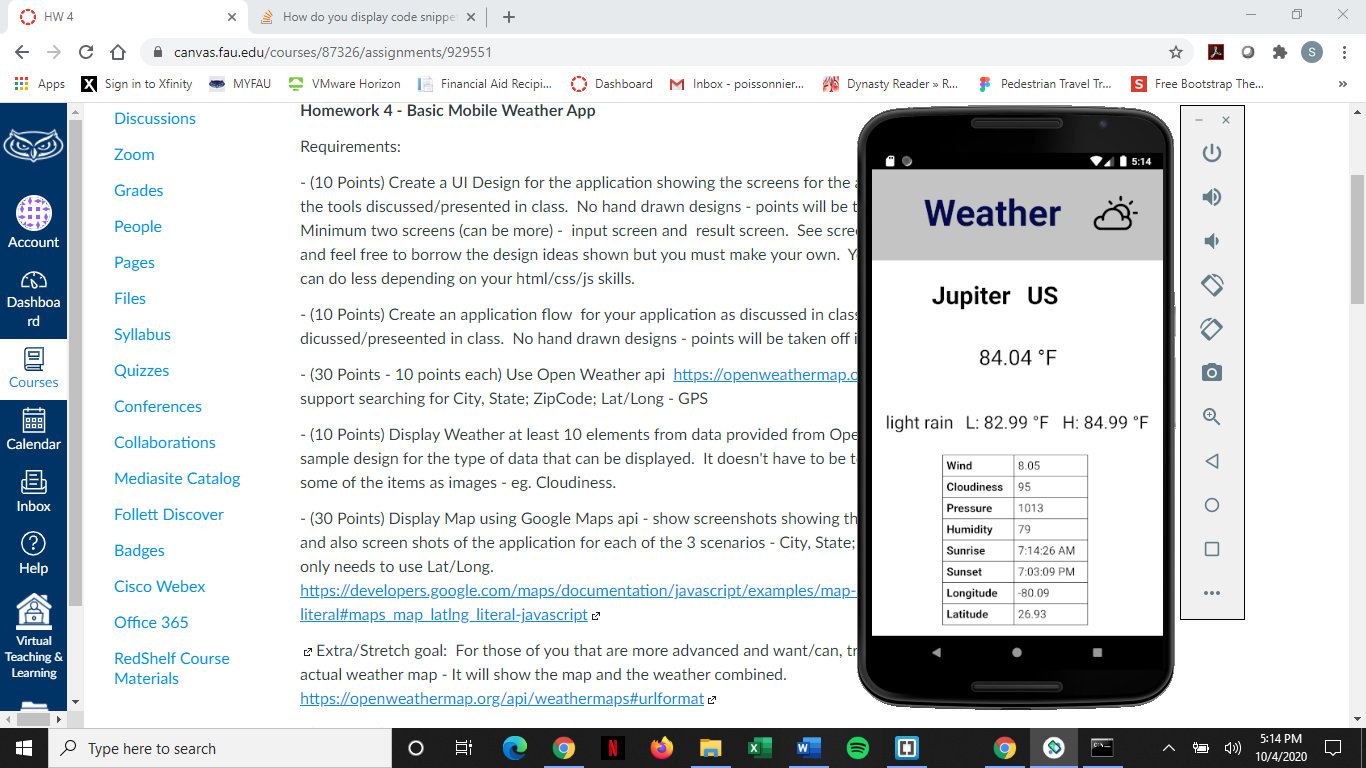
**Weather App**

****

****

****

****

****

**Index.html Code:**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no, minimum-scale=1.0, maximum-scale=1.0">

<link href="assets/css/styles.css" rel="stylesheet">

</head>

<body>

<div id ="Page1"><div class ='header'><img src="assets/pics/day+forecast+shine+sun+weather+icon-1320183298516188350.png" alt="Weather Symbol" style=" width:90px;height:90px;"><h1 class = "title">Weather</h1> </div>

<div class='search-view'>

<input type="text" placeholder="Enter City, State, or Zip Code" id ="term"/>

<button id ="submit-btn" onclick="return show('Page2','Page1');"></button>

<button id ="geo-btn" onclick="return show('Page2','Page1');"></button>

<p>Please enter City, State, ZipCode, or Click on the Location Button</p>

</div>

<footer><h4>Summer Poissonnier Z23492880</h4></footer>

</div>

<div id="Page2" style="display:none">

<div class ='header'><img src="assets/pics/day+forecast+shine+sun+weather+icon-1320183298516188350.png" alt="Weather Symbol" style=" width:90px;height:90px;"><h1 class = "title">Weather</h1> </div>

<h1><span id="name"></span> <span id ="country"></span></h1>

<p id="f"><span id ="temp"></span> °F</p>

<p id ="desc"></p>

<p id="fh">H: <span id ="tempmax"></span> °F</p>

<p id="fl">L: <span id ="tempmin"></span> °F</p>

<table style="width:50%" id="weather-data">

<tr>

<th>Wind</th>

<td id="wind"></td>

</tr>

<tr>

<th>Cloudiness</th>

<td id ="clouds"></td>

</tr>

<tr>

<th>Pressure</th>

<td id ="pressure"></td>

</tr>

<tr>

<th>Humidity</th>

<td id="humidity"></td>

</tr>

<tr>

<th>Sunrise</th>

<td id="sunrise"></td>

</tr>

<tr>

<th>Sunset</th>

<td id="sunset"></td>

</tr>

<tr>

<th>Longitude</th>

<td id ="lon"></td>

</tr>

<th>Latitude</th>

<td id ="lat"></td>

</table>

<iframe id ="map"

width="300"

height="400"

margin-left="30"

frameborder="0" style="border:0"

src="https://www.google.com/maps/embed/v1/place?key=AIzaSyDvrkPFr-kgIbdMVPFRON4Ox\_jqXLX830w

&q=center" >

</iframe>

<button class ="newsearch-btn" onclick="return show('Page1','Page2');">New Search</button>

<footer><h4>Summer Poissonnier Z23492880</h4></footer>

<script src="cordova.js"></script>

<script src="lib/jquery.js"></script>

<script src="js/index.js"></script>

<script src="js/weatherapp.js"></script>

<script src="js/app.js"></script>

</body>

</html>

**CSS Code:**

**.header**{

**background-color**: **#C4C4C4** ;

**text-align**: **center**;

**font-size**: **26px**;

**padding-bottom**: **.5px**;

**padding-top**: **.5px**;

**margin-bottom**:**30px**;

}

img{

**margin-top**: **30px**;

**margin-right**: **10px**;

**margin-left**: **90px**;

**float**:**right**;

**position**: **absolute**;

}

body{

**margin-top**: **0px**;

**margin-left**: **0px**;

**margin-right**: **0px**;

}

h1{

**color**:**#030A4e**;

**font-family**: "Arial Black", Gadget, **sans-serif**;

**margin-right**: **70px**;

}

**#term**{

**width**:**270px**;

**height**: **40px**;

**margin-left**: **30px**;

**margin-right**: **10px**;

**border**-radius: **6px**;

**font-size**: **21px**;

**font-family**:**inherit**;

}

**#submit-btn**

{

**height**: **28px**;

**width**:**28px**;

**font-family**:**inherit**;

**border**-radius: **6px**;

**background-image**: url(images/mag.png);

}

**#geo-btn**

{

**height**: **28px**;

**width**:**29px**;

**font-size**: **15px**;

**font-family**:**inherit**;

**border**-radius: **6px**;

**background-image**: url(images/NAVIGATIO-512.png);

}

**.search-view**{

**margin-top**: **100px**;

}

p{

**margin-top**: **60px**;

**margin-left**: **2px**;

**color**: cadetblue;

**text-align**: **center**;

**font-size**: **25px**;

**font-family**: **inherit**;

}

**#name**{

**text-align**: **center**;

**color**: black;

**margin-left**: **85px**;

**font-family**: **inherit**;

**font-size**: **35px**;

**margin-bottom**: **10px**;

}

**#country**{

**text-align**: **center**;

**color**: black;

**font-family**: **inherit**;

**font-size**: **35px**;

**margin-bottom**: **10px**;

**margin-left**: **15px**;

}

**#temp**{

**margin-top**: **0px**;

**margin-bottom**: **10px**;

**color**: black;

**font-size**: **30px**;

**text-align**: **center**;

}

**#f**{

**color**: black;

**font-size**: **30px**;

**margin-top**: **50px**;

**font-family**: **inherit**;

**margin-bottom**: **10px**;

}

**#desc**{

**margin-top**: **50px**;

**margin-left**: **20px**;

**float**: **left**;

**font-family**: **inherit**;

**color**:black;

}

**#fh**{

**float**: **right**;

**margin-top**:**50px**;

**margin-right**: **20px**;

**font-family**: **inherit**;

**color**:black;

}

**#fl**{

**color**:black;

**font-family**: **inherit**;

}

footer{

**font-family**: **inherit**;

**margin-left**: **90px**;

**color**: black;

}

**#page2**{

**overflow**: **scroll**;

}

table, th, td {

**border**: **1px** **solid** black;

**border-collapse**: **collapse**;

}

th, td {

**padding**: **5px**;

**text-align**: **left**;

}

**.table-view** {

**margin**: **0** !important;

**border**: **none**;

}

h4{

**font-family**: **inherit**;

**margin-top**: **300px**;

**color**: black;

}

**#weather-data**{

**margin-top**: **30px**;

**margin-left**:**100px**;

**margin-bottom**: **30px**;

}

**#map**{

**margin-left**: **50px**;

}

**.newsearch-btn**{

**margin-top**:**30px**;

**margin-left**: **160px**;

**height**: **30px**;

**font-size**: **15px**;

**font-family**:**inherit**;

**border**-radius: **6px**;

}

**Weatherapp.js Code:**

**var** OpenWeatherAppKey = "7e59cb29b6de97ff758260f346d64075";

**function** getWeatherWithZipCode() {

**var** zipcode = $('#term').val();

**var** queryString =

'http://api.openweathermap.org/data/2.5/weather?zip='

+ zipcode + ',us&appid=' + OpenWeatherAppKey + '&units=imperial';

$.getJSON(queryString, **function** (results) {

showWeatherData(results);

}).fail(**function** (jqXHR) {

$('#error-msg').show();

$('#error-msg').text("Error retrieving data. " + jqXHR.statusText);

});

**return** **false**;

}

**function** getWeatherWithCity() {

**var** city = $('#term').val();

**var** queryString =

'http://api.openweathermap.org/data/2.5/weather?q='

+ city + ',us&appid=' + OpenWeatherAppKey + '&units=imperial';

$.getJSON(queryString, **function** (results) {

showWeatherData(results);

}).fail(**function** (jqXHR) {

$('#error-msg').show();

$('#error-msg').text("Error retrieving data. " + jqXHR.statusText);

});

**return** **false**;

}

**function** showWeatherData(results) {

**if** (results.weather.length) {

$('#error-msg').hide();

$('#weather-data').show();

$('#name').text(results.name);

$('#country').text(results.sys.country);

$('#temp').text(results.main.temp)

$('#wind').text(results.wind.speed);

$('#humidity').text(results.main.humidity);

$('#lon').text(results.coord.lon);

$('#lat').text(results.coord.lat);

$('#desc').text(results.weather[**0**].description);

$('#tempmax').text(results.main.temp\_max);

$('#tempmin').text(results.main.temp\_min);

$('#pressure').text(results.main.pressure);

$('#clouds').text(results.clouds.all);

**var** sunriseDate = **new** Date(results.sys.sunrise \* **1000**);

$('#sunrise').text(sunriseDate.toLocaleTimeString());

**var** sunsetDate = **new** Date(results.sys.sunset \* **1000**);

$('#sunset').text(sunsetDate.toLocaleTimeString());

} **else** {

$('#weather-data').hide();

$('#error-msg').show();

$('#error-msg').text("Error retrieving data. ");

}

}

$('#submit-btn').click(getWeatherWithZipCode);

$('#submit-btn').click(getWeatherWithCity);

$('#geo-btn').click(getWeatherWithGeoLocation);

**function** getWeatherWithGeoLocation() {

navigator.geolocation.getCurrentPosition(onGetLocationSuccess, onGetLocationError,

{ enableHighAccuracy: **true** });

$('#error-msg').show();

$('#error-msg').text('Determining your current location ...');

$('#geo-btn').prop('disabled', **true**);

}

**function** onGetLocationSuccess(position) {

**var** latitude = position.coords.latitude;

**var** longitude = position.coords.longitude;

**var** queryString =

'http://api.openweathermap.org/data/2.5/weather?lat='

+ latitude + '&lon=' + longitude + '&appid=' + OpenWeatherAppKey + '&units=imperial';

$('#geo-btn').prop('disabled', **false**);

$.getJSON(queryString, **function** (results) {

showWeatherData(results);

}).fail(**function** (jqXHR) {

$('#error-msg').show();

$('#error-msg').text("Error retrieving data. " + jqXHR.statusText);

});

}

**function** onGetLocationError(error) {

$('#error-msg').text('Error getting location');

$('#geo-btn').prop('disabled', **false**);

}