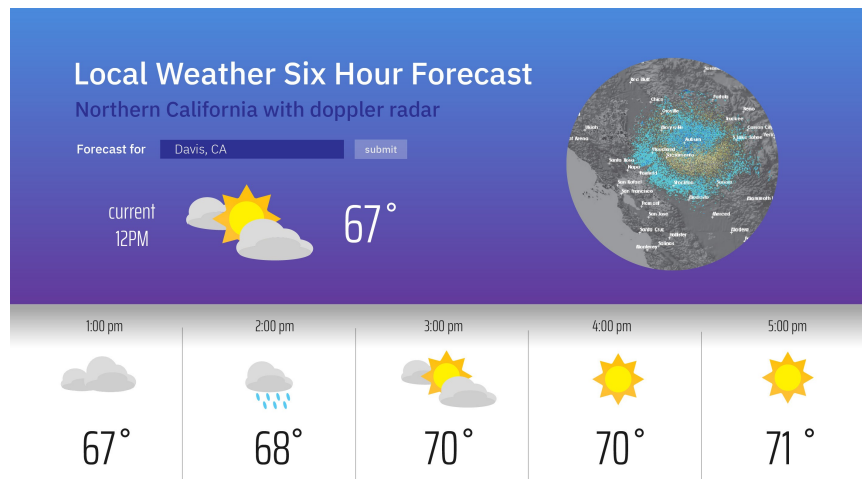


## ECS 162 Assignment 3, Spring 2019

Please submit by 10pm, Thurs April 25,  
using Canvas.

You may do this assignment with a  
partner.



We will use the [OpenWeatherMap Hourly Forecast API](#) to make a weather widget. By default, our app shows the weather in Davis, but the user can select other cities in the region as well. Current conditions appear on the left, and a doppler radar map from the [National Weather Service](#) on the right; the map can be animated. Along the bottom, we see the weather forecast for the next six hours.

Entering a city name or a zip code in the box on the left should either switch to showing the weather for that place, or leave the display the same and put up a message saying "Not found". Since the radar map is roughly a 150-mile radius around Sacramento,

places farther away than that can be "Not found".

Our main goal in this assignment will be using the OpenWeatherMap API to get the forecasts and getting that information onto the screen. Animating the weather map is the second most important part. The third priority will be getting the display to look like the design, as nearly as possible; we'll learn some new css in the process.

I recommend you stick to pure Javascript and CSS (including flexbox), avoiding JQuery, and other libraries, since that is what will be on the midterm.

## Forecasts

The OpenWeatherMap Hourly Forecast API distributed information, in JSON, using the CORS protocol. The result returned by OpenWeatherMap includes a JSON string, from which we can extract the current conditions and forecasts for the next six hours. To use the API, you need an API key; sign up on their Web page. The URL to request the Davis hourly forecast looks like this:

```
http://api.openweathermap.org/data/2.5/forecast  
/hourly?q=Davis,CA,US&units=imperial&APPID=xxx
```

where the xxx is replaced by your API key. This demo, [makeRequest.zip](#), shows how to make a CORS request to OpenWeatherMap; put your API key into the URL in order to try it out. Or, you can just put the URL into your browser and see what happens.

## Doppler Radar Map

The doppler radar map is downloaded from the National Weather Service. It is composed of three layers, each of which is an image: the topographic map, which is opaque, the city names, which is transparent except for the text, and the actual doppler radar, which changes every 10 minutes. The URLs for these images are:

```
http://radar.weather.gov/ridge/Overlays/Topo/Short  
/DAX_Topo_Short.jpg  
http://radar.weather.gov/ridge/Overlays/Cities/Short  
/DAX_City_Short.gif
```

[https://radar.weather.gov/ridge/RadarImg/N0R/DAX\\_N0R\\_0.gif](https://radar.weather.gov/ridge/RadarImg/N0R/DAX_N0R_0.gif)

To make the display, the cities and doppler have to be overlayed over the topographic map; now is the time to try "position: absolute". To put the map into its circular frame, we can use the css [clip-path](#) property.

To animate the doppler radar map, we need not one doppler image but several. Here is a demo [collectPastDoppler.zip](#), which gets the ten most recent doppler radar overlays from the National Weather Service directory. To cycle through the ten images, I suggest piling them all onto the weather map with "display: none", and then, one by one, changing to "display: inline".

## Design

This archive [weatherDesign.zip](#) contains mock-ups for tablet and Web views. There are two versions of the mobile mockup, one for when it loads and one for after the up-arrow is pressed (down-arrow goes back to the first view). Use a [css animation](#) to get a smooth "slide up/slide down" transition between the two views. The appStandards file show color and font specs. The "assets" directory contains 11 SVG icons for the different weather conditions. OpenWeatherMap gives an icon code with every forecast, and they have a day version and a night version of each icon. For icon codes where we don't have a night version, just use the day version.

## Turn in Instructions

Hand in the assignment as a zip file, using Canvas. You can include the assets (feel free to change them). Be sure to check that unzipping gives you a working app, with nothing missing. If you work with a partner, both of you should turn in the same code, and include a readme with both your names on it.