Moon Phase and Weather Information

*Mashups Final Project Documentation*

Muhammad Rafay Ashfaq

Abstract:

This is the documentation of the *Moon Phase and Weather Information* interactive web application developed for Mashups, an interactive media course on web development.

This application successfully interfaces between the user and 2 separate APIs to provide an interactive experience of viewing the current moon phase and weather information of any city requested.

Concept:

The inspiration behind the application was my love for timekeeping. I find celestial objects and their motion to be very interesting and thus decided to build an application that had something to do with them. Initially, I wanted to make an application with which you can track the motion of different celestial objects: stars, galaxies, planets and asteroids. After researching ways of doing so, I realized that this application’s intensity exceeds the scope of this course. Additionally, APIs that allow one to track the position of all celestial objects do not exist. The complexity required would not have been suitable for a single-front and simple application like the prompt for the assignment at hand. Following this realization, I decided to narrow my focus down to one celestial object: the moon.



Figure 1 The moon

Our mysterious moon has been the subject of interest since the beginning of humanity. This astronomical body orbits earth as its only natural satellite. It is thought to have formed 41.5 billion years ago, shortly after the formation of earth. A widely accepted view is that the moon formed as the debris of a collision between the earth and a Mars-sized body called Theia. The moon orbits the earth 384,402 km away, synchronously with the earth thus we only see one side of it on Earth. Fig.1 shows the texture of the moon which is thought to be made of dark volcanic material. The different phases of the moon are shown below in Fig. 2

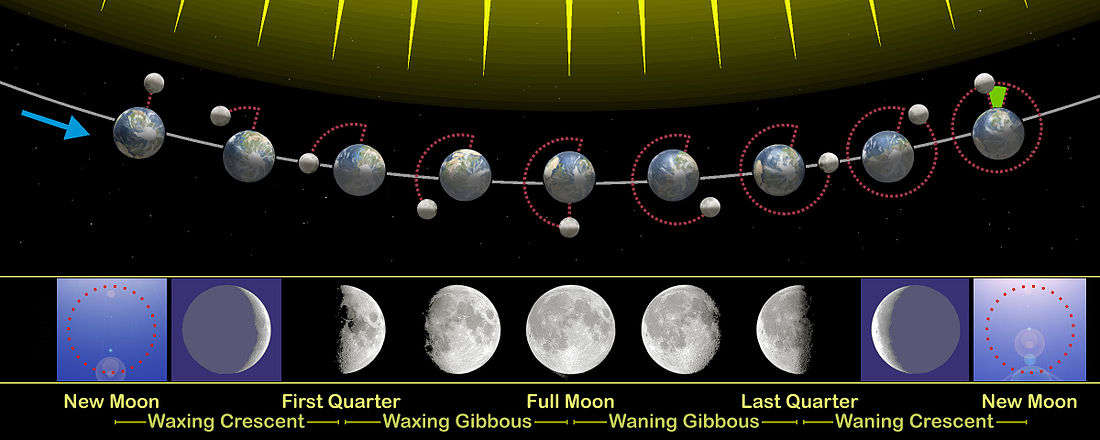


Figure 2 Moon phases

I wanted the app to surely show the beautiful and different phases of the moon. As most people do not know the different phases of the moon and the sequence in which they appear, I also wanted this application to have an educational feature to educate people on the particular moon phase of that point in time.

I also thought it would be useful to use another API to give weather information for the day. This led to the concept of the app to be : An app that gives you the moon phase and weather information of a particular city. The city name could be manually entered, otherwise it would be manual by default.

Process

The first step was to find the required APIs that I can use to obtain information. I discovered an exceptional API developed and hosted by the US Naval Observatory. The API gave all kinds of detailed information about the moon. Initially, I picked this API but in the middle of the development process, the API stopped responding, after which I emailed the relevant authority. A screenshot of the conversation is shown below.

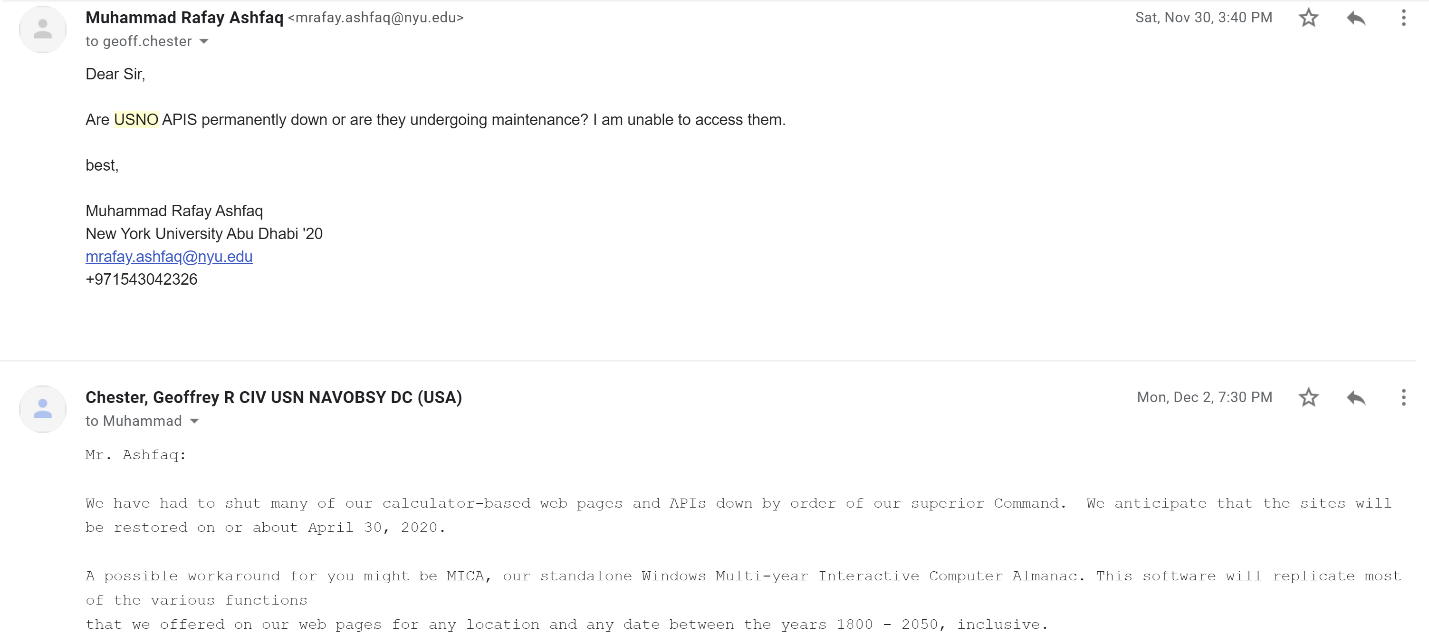


Figure 3 Conversation with US Navy

The entire application had been designed around this API but it was shut down without notice. I had to expand my search to find other APIs and decided to use world weather online APIs. They had a weather and astronomy API that suited my purpose.

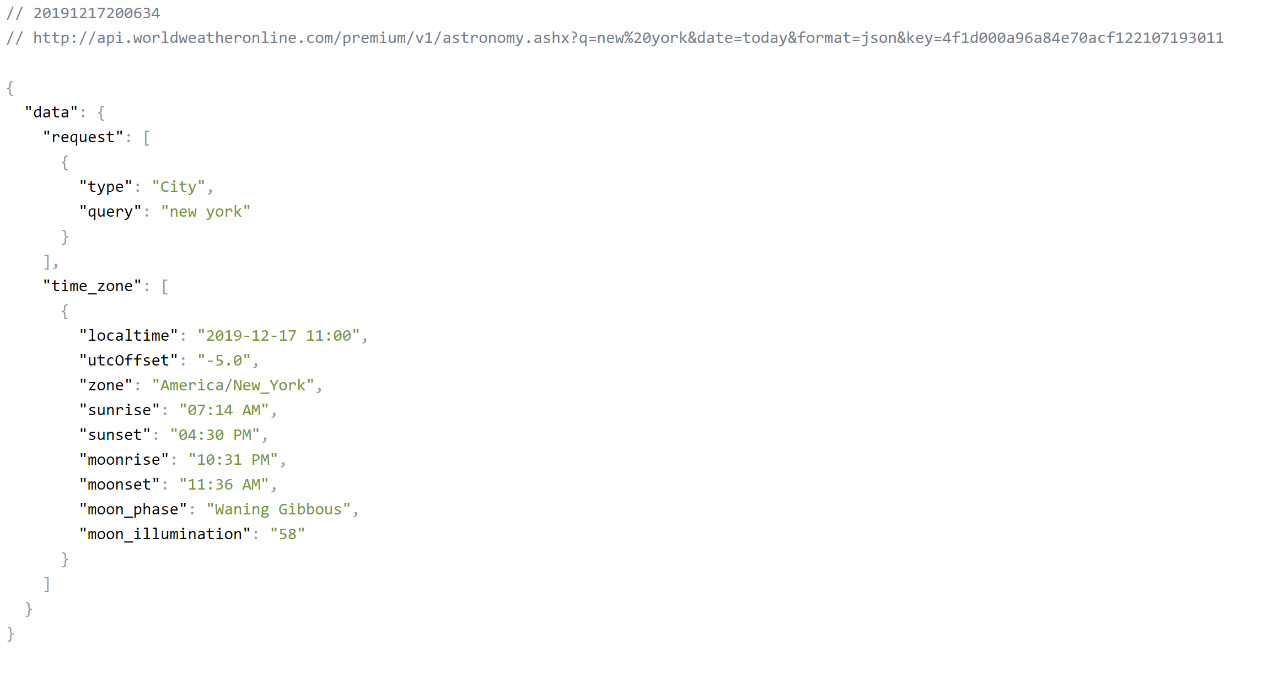


Figure 4 Astronomy API

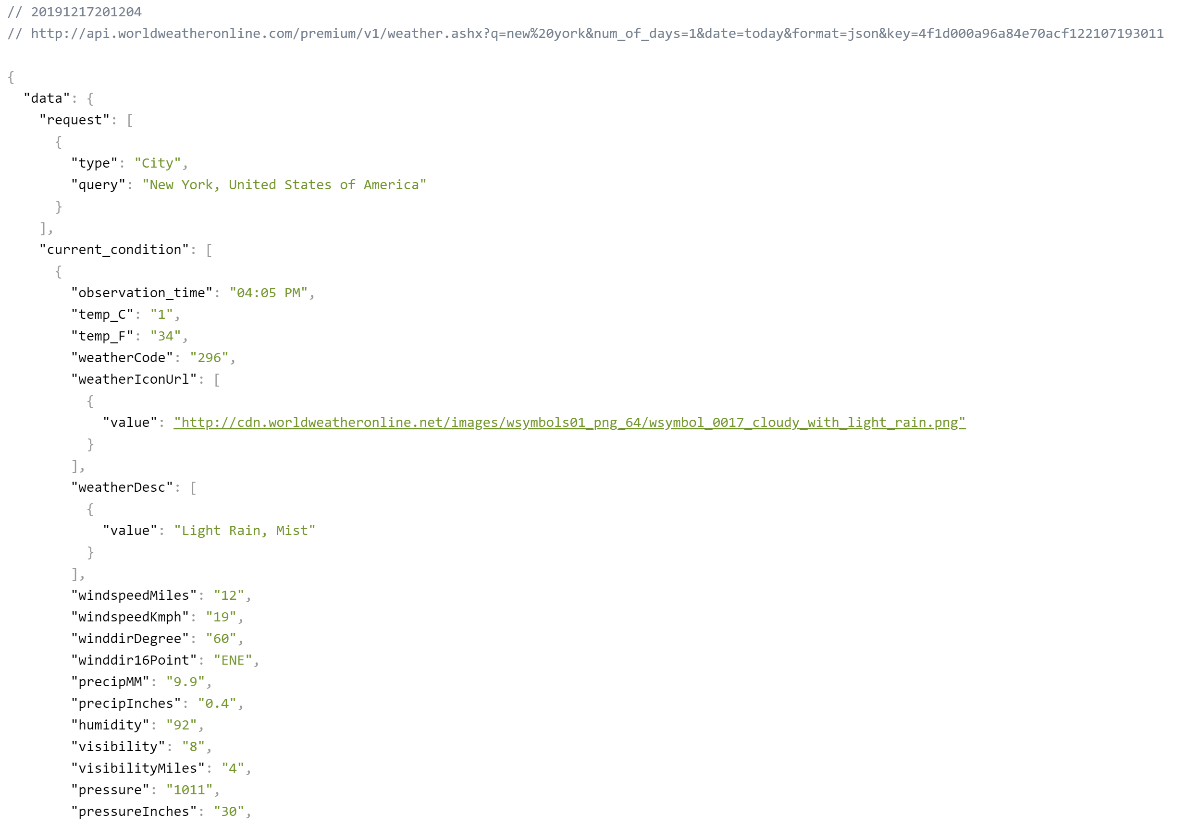


Figure 5 Weather API

The screenshots show the information obtained from the APIs.

The application was decided to be developed using P5.js due to its extraordinary visual design features. The lerp color function was used to make a gradient in the background for aesthetic purposes. A particle system using object oriented programming was used to create the twinkling stars affect.The httpget feature was used to pull information from the APIs. The final result looked like this.

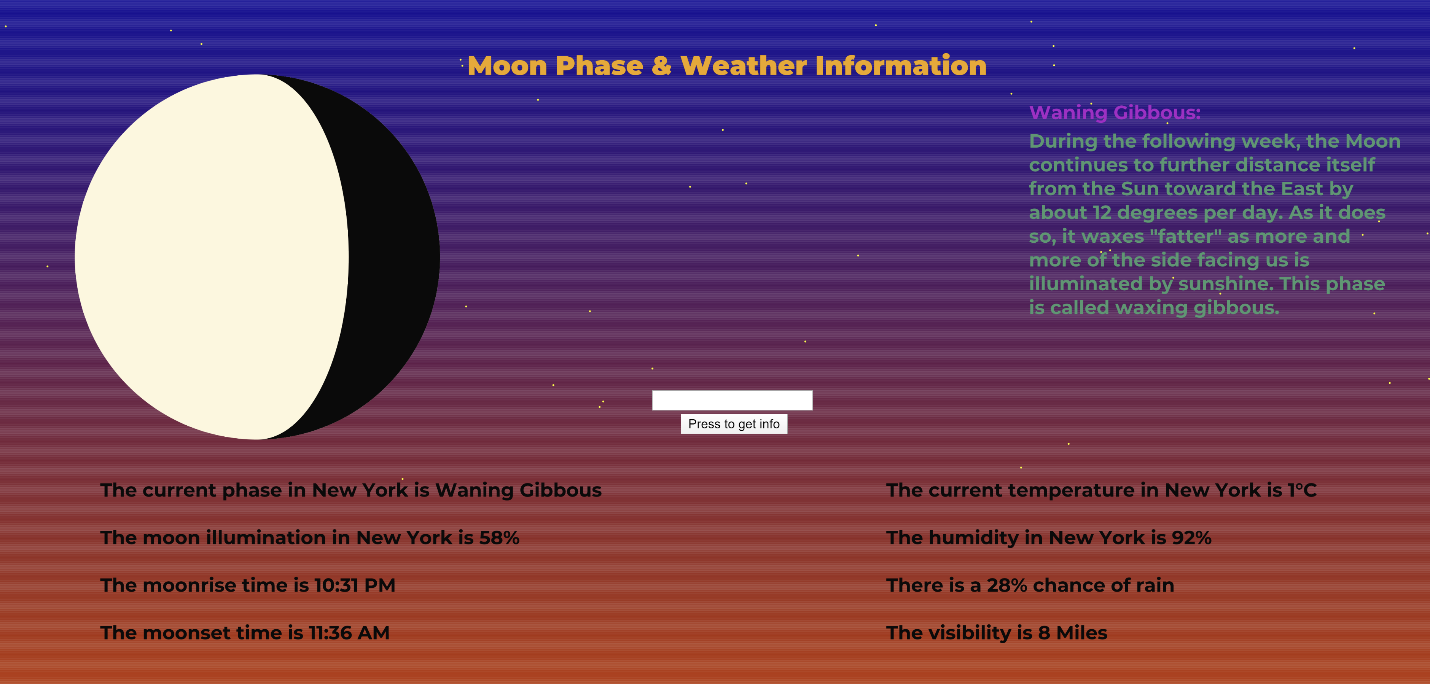


Figure 6 Final working project

The app successfully

1. Incorporates user input
2. Handles error
3. Prints moon phase
4. Prints moonrise time
5. Prints moonset time
6. Prints moon illumination percentage
7. Gives current temperature
8. Gives chance of rain
9. Gives information the moon phase
10. Gives visibility information for astronomy and safety

Problems

The biggest problem I faced was dealing with a CORS error shown in Fig. 7.

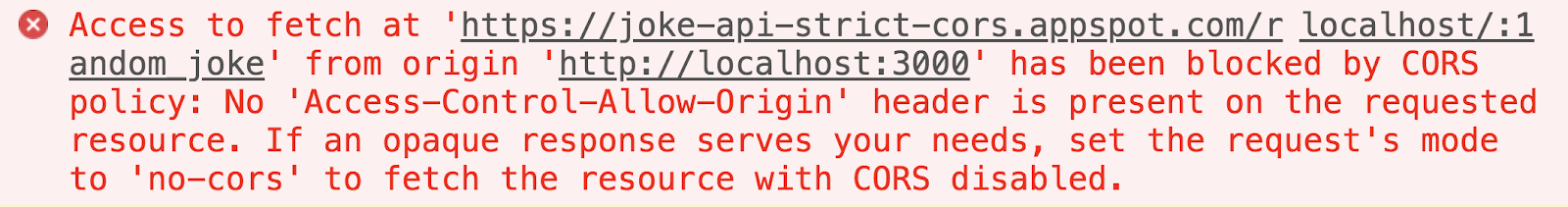


Figure 7 CORS Error

The error stems from a security mechanism that browsers implement called the same-origin policy. The same-origin policy fights one of the most common cyber-attacks out there: cross-site request forgery. In this maneuver, a malicious website attempts to take advantage of the browser’s cookie storage system. For every HTTP request to a domain, the browser attaches any HTTP cookies associated with that domain. This is especially useful for authentication.

Browsers prevents the API from getting special access to cookies in this way to keep your data secure and stop phishing attempts.

To solve this problem, I used JSONP which is a different authentication process that uses a callback and is therefore much safer.

User Testing

After user testing I discovered that the website does not handle errors well and so error handling features using try-catch were added to print an error incase the wrong city name was entered. This feature works successfully as shown in the screenshot below.

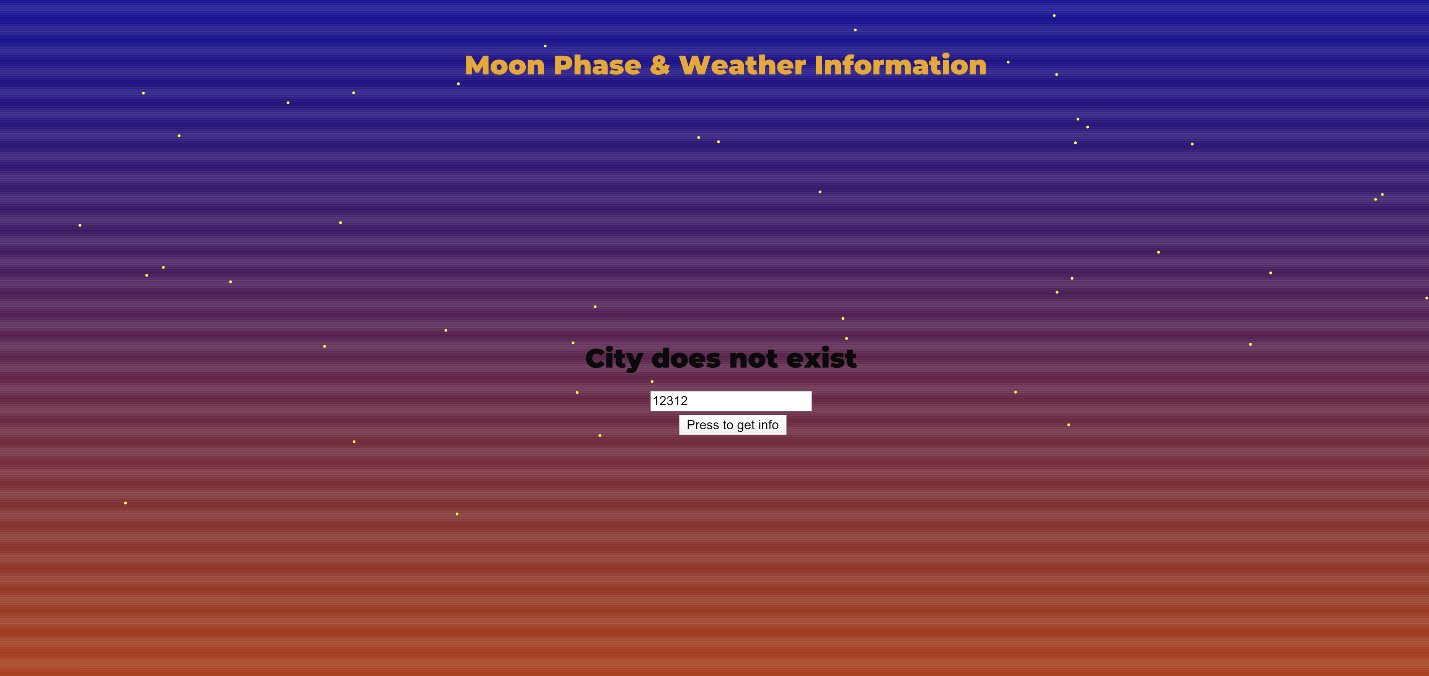


Figure 8 Error handling

As the figure shows, the site properly prints an error message and prevents the moon from being displayed.

The application also successfully takes user input to show the correct information, and by default prints New York’s details.

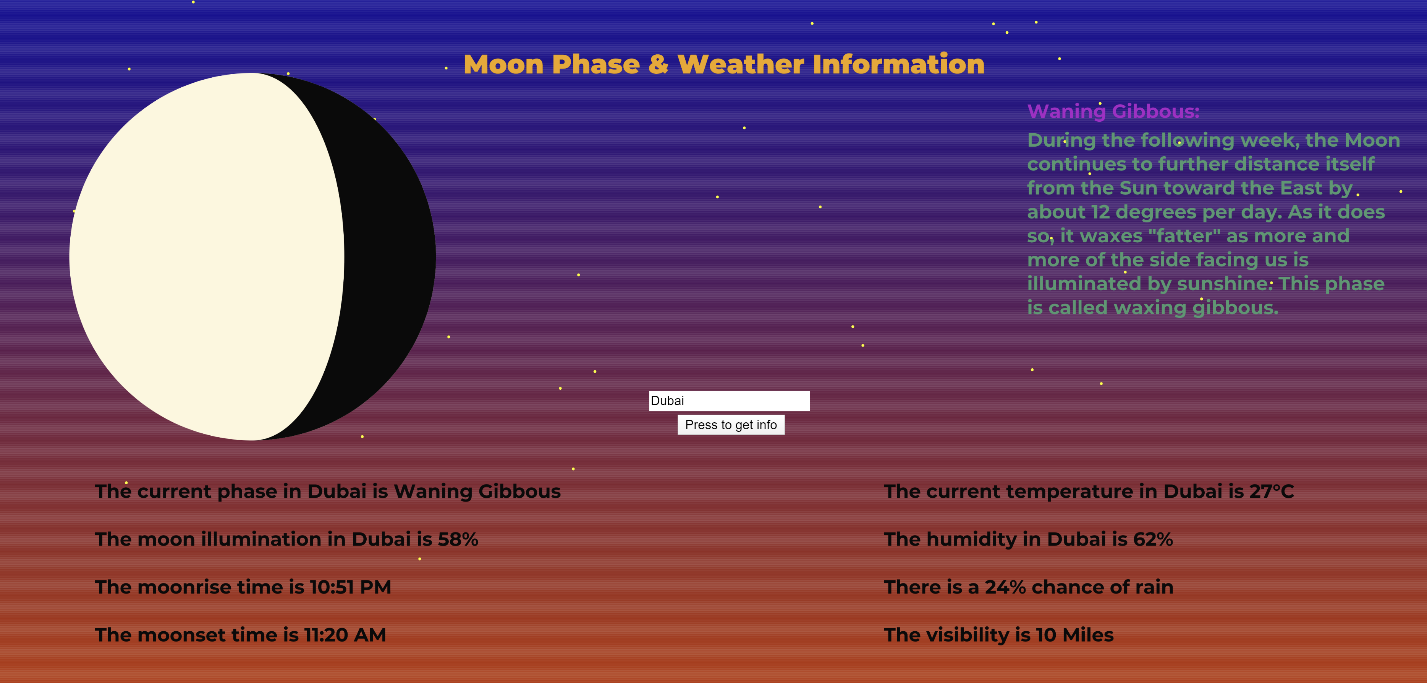


Figure 9 Successful incorporation of user input