

Module 4 – Project Code

Use the space below to enter the full URL for your project code, hosted on GitHub. Please ensure that the URL you enter is properly hyperlinked (use the Insert>Hyperlink menu item). If you copy and paste the URL directly from a browser it should automatically be hyperlinked.

<https://github.com/Iliyan-Darediya/dgl-213-Module-4>

Module 4 – Documentation

Use the spaces below to summarize: 1. The hours spent on the *coding* portion of this module project; 2. Parts of your code that do not work as expected, or that are incomplete. You may write using bullet points, rather than complete paragraphs, as long as each bullet point is written as a complete sentence.

Approximate hours	8
Bugs or incomplete code	<ol style="list-style-type: none">1. Re-entering the value in the movie name after executing a query for previous query, doesn't clear the cards for the previous search.2. Using movie names that has a space doesn't provide the expected result

Module 4 – Reflection

Use the spaces below to answer the questions posed.

Module-specific questions

1. In what way did the data available at your chosen API endpoint influence the design of your app?

I had to reconsider my application. For milestone zero I wished to have a search by movie's genre. But the new york times api didn't have valid endpoints to execute this. Thereby I had to re-focus my vision towards a page that contains the data about the critics.

2. What code-based compromises were you forced to make during development as a consequence of the API you chose? If you were not forced to make any compromises describe a part of your app that you think works extremely well and explain why.

Ideally I'd use the animejs library in a different file. My plan was to write all the functionality of the animejs file in a animation.js file. But this wasn't possible, as I was using an async function to add data to the dom. So I had to re-write the code that influences the animation in two separate files(name.js and critics.js)

3. In what way did the functionality of your chosen library influence the design of your app?

Animejs didn't diverge my mind from my chosen look for the application. Rather it helped me in my design. I was looking at Three.js, but it seemed that it wouldn't let me make changes to my individual elements, animejs gave me the power that three didn't possess.

4. What code-based compromises were you forced to make during development as a consequence of the library you chose? If you were not forced to make any compromises describe a part of your app that you think works extremely well and explain why.

Ideally I'd just use npm install to add the required dependencies for my library, but doing so didn't work out well as there were mixed information of the library versions. So I had to manually download the file anime.min.js and add it to my repository. Doing so was the only way I could execute the code that I had in mind for the animation. So because of the mixed documentation, I had to compromise the organisation of my code.

Process questions

1. What did you learn in this module that will be most valuable to you as you progress through your education? What was most valuable to your future career?

I learnt how to read documentation of chosen libraries. I also learnt how we can make our projects compatible in accordance with the data available on api. But most importantly I'd say that learning to divert is the best skill that I learnt in this module. As a developer I won't have access to all the information at all times. It's upto me to make the best out of the provided data

2. What was most challenging aspect of this module? What strategies did you use to overcome these challenges?

The most challenging aspect of this module was the implementation of the animejs library. The reason behind this was that there were a lot of different versions out there and each version has upgrades over the other. But sadly the developers didn't update the documentation most efficiently. That's the reason it took a bit of extra time and effort. I overcame this issue by following threads on stackoverflow and reddit. Past users were helping out the new developers. This was a great sight to behold.