Fullstack Engineer Coding Challenge

Objectives

This coding challenge contains two questions pertaining to fullstack development. Question 1 will test your knowledge with JSON, REST, and file systems. Question 2 will test your background with web components, JavaScript, and CSS. It should take you around 2 hours of your time to complete.

Rules of Submission

1. This challenge must be completed in under 24 hours.
2. Your code and any assets must be version controlled with Git. In the email, you must provide a hyperlink to a github.com repository that you have pushed your project to.
3. The github project should have a README file that includes instructions on anything that would need to be done to **run** your submission.
4. Submit your challenge by sending an email to codechallenge@lightfeather.io

Question 1 - Webserver

Create and document a webserver that has all the capabilities requested below. Please use whatever programming language you think is best for the job

Webserver Specifications

This webserver will provide the user the ability to encode messages with a shift cipher.

- ❏ The webserver should run on port 23456.
- ❏ The webserver has an endpoint at /api/encode.
    - ❏ This endpoint should accept post requests.
    - ❏ This endpoint expects a JSON string to be received. The JSON structure should have two keys.
        - ❏ The JSON structure should have a key Shift. It's value should be an integer.
        - ❏ The JSON structure should have a key Message. It's value should be a string.
    - ❏ When the JSON payload is received, it should encode the provided message using the shift cipher method. If multiple words are provided, spaces should be retained.
        - The shift cipher works by shifting each letter a given number of letters to the right in alphabet order. The letter A shifted 1 would be B. B shifted 2 would be D.
        - If Shift = 3 and Message = "dad", the encoded message would be generated as "gdg".

    - If Shift = 2 and Message = “the eagle has landed”, the encoded message would be generated as “vjg gcing jcu ncpfgf”.
    - This should work for any positive shift value provided. If Shift = 522 and Message = “the eagle has landed”, the encoded message would be generated as “vjg gcing jcu ncpfgf”.
- ❏ When the message is successfully encoded, it should be stored in a file on disk.
- ❏ The endpoint should return a 200 if successful, 500 if unsuccessful.
- ❏ The endpoint should return a json object with a single key EncodedMessage. The value returned should be the string you successfully encoded. If unsuccessful, it should return an empty string.

Question 2 - Web Component

Please create the web component of a signup page that is typically seen on web apps. Feel free to use any version of JavaScript and/or your framework/library of choice.

Here is an example: https://dashboard.stripe.com/register

Web Component Specifications

- Create a user signup that accepts the following fields with respective labels:
    - Username
    - Email
    - Password
    - Confirm Password

- Ensure the following prerequisites:
    - Make sure a username cannot be longer than 15 characters.
    - Validate the email addresses to spec: https://stackoverflow.com/a/2049510
    - Confirm the ‘Password’ and ‘Confirm Password’ fields match.

- Please show an error when the user clicks outside the field and the info entered is incorrect.
- Create submit button that is disabled until all requirements are met.

- Follow the following style rules:
    - Align the fields vertically in a ‘stack’ like format.
    - Center the input fields on the page.
    - Make sure there is at least 8px of vertical margin between each field. (and Submit button)
    - Give the input fields rounded corners of at least 2px.