Intro

You are tasked with building a credit card number validation service. There are a number of key objectives that you will need to put together along with a stretch task if you wish to try. The stretch goal is not required but would provide more insight into your understanding of containers, or learning capabilities if the task is new to you. Please upload your solution to github and share a link with us.

The Service

Build a service that can validate credit card numbers using an http get request. The below points detail the task:

- you may use NodeJS backend to build the service using Express or NestJS
- create a an api endpoint in your service which can be called using an HTTP GET. The path should be `/api/validate/${creditCardNumber}`. Example curl:

```
curl http://localhost:myport/api/validate/49927398716
```

- if the creditCardNumber is a numeric string return an HTTP 200 response with a JSON body as below where isValid indicates if it is a valid credit card number:

```
{
    "isValid": true/false
}
```

- if the creditCardNumber is not a numeric string (e.g. `a54g65`) return a 400 HTTP response with a JSON body as below:

```
{
    "error": "Credit card number must be numeric"
}
```

- the validation of the credit card numbers should be done using Luhns Algorithm. Use the below psuedo code for the implementation:

```
function isAValidCreditCard(cardNumber: string)
   sum := 0
   parity := length mod 2
   # loop backwards through digits of the card
   for i from (length − 1) to 0 do
      currentDigit = cardNumber[i] as integer
      if (i + 1) mod 2 == parity then
         sum := sum + currentDigit
      elseif currentDigit > 4 then
         sum := sum + 2*currentDigit − 9
      else
         sum := sum + 2*currentDigit
```

```
        end if
    end for
    return sum mod 10 == 0
end function
```

- write unit tests for the `isAValidCreditCard` function above which will verify it returns the below results for the given credit card numbers:

Credit Card Number	**Is Valid**
49927398716	True
49927398717	False
1234567812345678	False
1234567812345670	True
2222405343248877	True
2222990905257051	True

UI

Create a simple React App which has a screen with an input box. Inputting text should call the endpoint exposed from the service.

Containerising the Application

Add a dockerfile that will package and run the application. It should be buildable and runnable using the below commands:

```
docker build . -t creditcardvalidator:latest
docker run -it -p 8080:your_apps_port creditcardvalidator:latest
```

CI

Add a simple github action to your repo which will do the following:

- run on pushes to your default branch (`main` or `master` for example)
- will run an installation of your apps dependencies
- will check if your code compiles
- will run your unit tests

Extension task (optional)

Put together instructions to run this application using docker compose or a local kubernetes installation.

Useful Links

A good place to start with Github actions are the "starter workflows" in each language specific section here: https://docs.github.com/en/actions/automating-builds-and-tests

You can read about Luhns algorithm here if you are interested but understanding this algorthim is not important: https://en.wikipedia.org/wiki/Luhn_algorithm