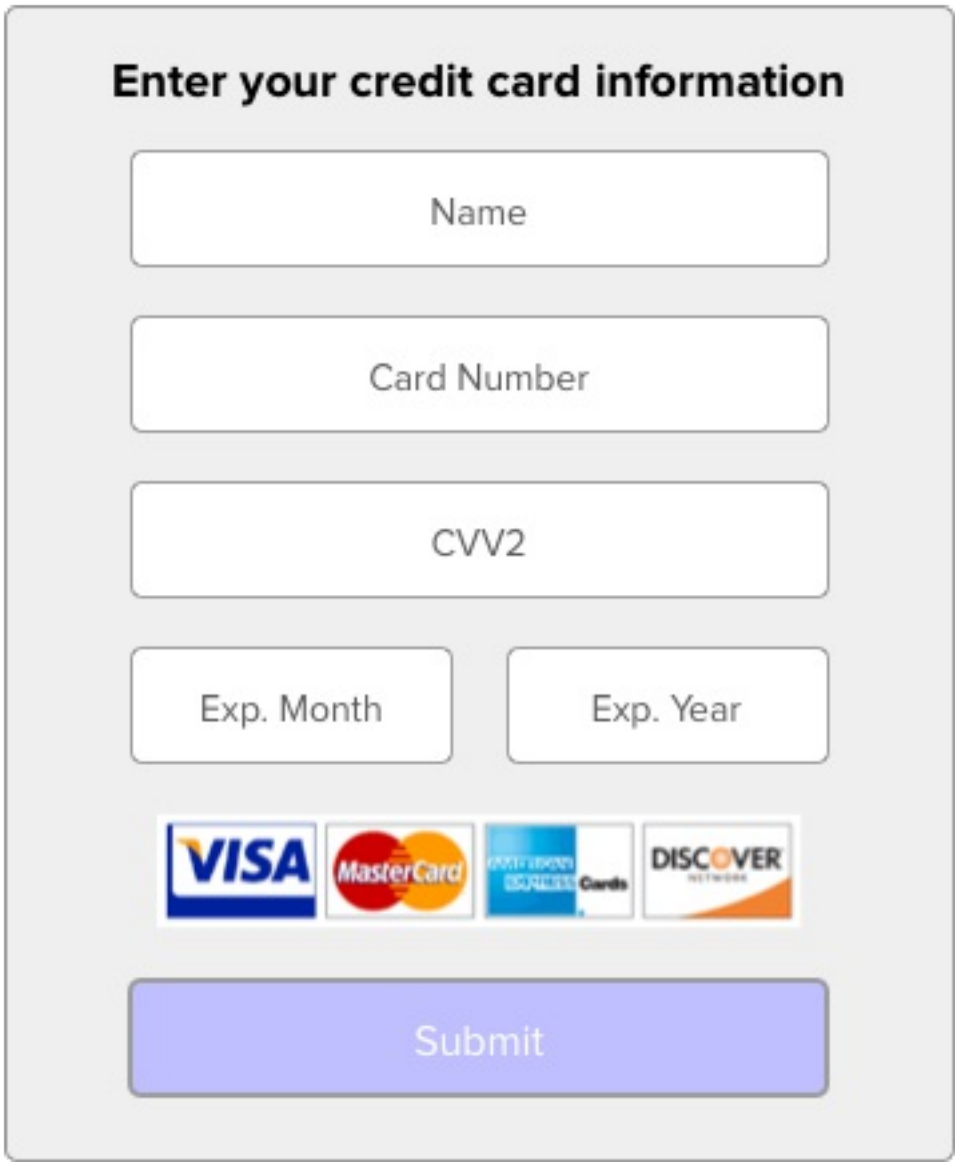
Affirm Take Home Front End Coding Challenge

Overview

Design and build an input component that lets a user submit credit card information, with considerations on user experience and validation. We encourage you to think through how a user would interact with your UI before implementing it in code.

Design Wireframe



Inputs

The user should be able to enter these fields:

- Name

- Credit card number
- CVV2
- Expiration date
    - Month
    - Year

There are a handful of established standards credit card numbers. For simplicity, we'll only consider Visa and AmEx's:

- Visa
    - Credit card numbers
        - 4 groups of 4 numbers
        - 16 characters total
        - Start with "4"
    - CVV2
        - 3 characters
- AmEx
    - Credit card numbers
        - 4-6-5 number grouping
        - 15 characters total
        - Start with "34" or "37"
    - CVV2
        - 4 characters

Expiration dates are considered valid if they're set after the current month and year.

Keeps these constraints in mind when checking for user input validity.

Deliverables

Implement a webpage (index.html) that renders a UI allowing for users to enter their credit card information. Ideally, your solution should satisfy following criteria:

1. The solution should be implemented using one of the modern Javascript frameworks (We use React here at Affirm).
2. The interface should provide enough validation capabilities. If there is a validation error, your UI should let the user know where it goes wrong.
3. The interface should be properly styled. The wireframe provides a basic idea of the desired look, but feel free to be creative and refine the design.
4. There should be basic test coverage of your code.

Beyond these basic requirements, feel free to explore interface features that manipulate the user's inputs to reduce the occurrence of errors or provide easier understanding of input errors. Conversely, also think about what you can

do to encourage users that they're on the right path while entering long strings of digits.

You should also include a write-up that answer these questions:

1. How long did you spend working on the problem? How much time did you spend thinking about the design before writing your code?
2. What are the UI/UX usability features you implemented, or thought about implementing? How do they help validate the user input?
3. What would a form submission/API payload of this look like? How would you deal with validation errors that may come from that API response?
4. What are some styling and layout considerations for these types of form inputs?

Feel free to include any additional comments in your write-up.

You should send us a folder containing the source code to your program as well as the above write-up.

Evaluation

Your project will be evaluated based on the following criteria, in decreasing order of importance:

1. *Correctness* - Can the user enter credit card information, and be shown an error if their input fails to validate?
2. *Clarity* - Is the code well-organized and easy to read? Is it well-tested?
3. *Usability* - Is the UI styled properly? Is it easy to make mistakes when entering information? Does the interface help correct mistakes?
4. *Extensibility* - Can this component be reused and dropped onto a page? Is it easy to add additional credit card types and their constraints?

Final Notes and Suggestions

- We suggest starting with a simple, straight-forward interface that meets the requirements first, then to iterate on user experience improvements.
- It's not necessary to support multiple browsers.
- Feel free to use any language (e.g., one that cross compiles to JavaScript), libraries, frameworks, or tools that makes your job easier. We suggest to not go overboard with front end tooling; consider the problem domain and how your choice of tools may affect the portability and reusability of your code.
- If you have time after building a component, feel free to express your skills and creativity with styling and design choices. We will not penalize solutions for sticking with barebones design, but it is an area to showcase your front end abilities.