Write an angular application with the following requirements:

1. Create DTO for modeling Credit Card Payment details like below, which will be used to make requests
    a. Credit Card Number (mandatory, string)
    b. Card Holder (mandatory, string)
    c. Expiration Date (mandatory, Date, >CurrentDate)
    d. Security Code - CCV (optional, string, 3 digits)
    e. Amount (mandatory, number, > 0)
2. Using NgRx, create a state management solution that will hold our card.
3. Write a Payment service with a function that creates a POST request.
4. Create a new page and a new component (to be used in this page), with inputs for the DTO, created at point 1, use reactive forms and add validations on these inputs. Create a button with a click event and call the payment service method and based on the http response, show a toast notification informing the user.
5. In the app.component.html, create a button (name it any way you like) and use the Angular router to navigate to the new page created at the previous point.
6. To make sure that our state management solution is working, get the data from the store and display it on the app.component.html.

Obligations:

- Use the angular/typescript style guide to separate the models/dto and the services.

- Use RxJS when making requests and demonstrate some operators.

- Make sure to avoid any memory leaks.

- Make sure you have a good file hierarchy and follow best practices to improve naming conventions

Recommendations:

- Follow clean code best practices, follow industry standards (https://angular.io/guide/styleguide)

- Use CSS or SCSS to style your component(s). Although functionality is more important, you can also make it look pretty.