SPRIBE

Test Assignment for the Angular Developer Position

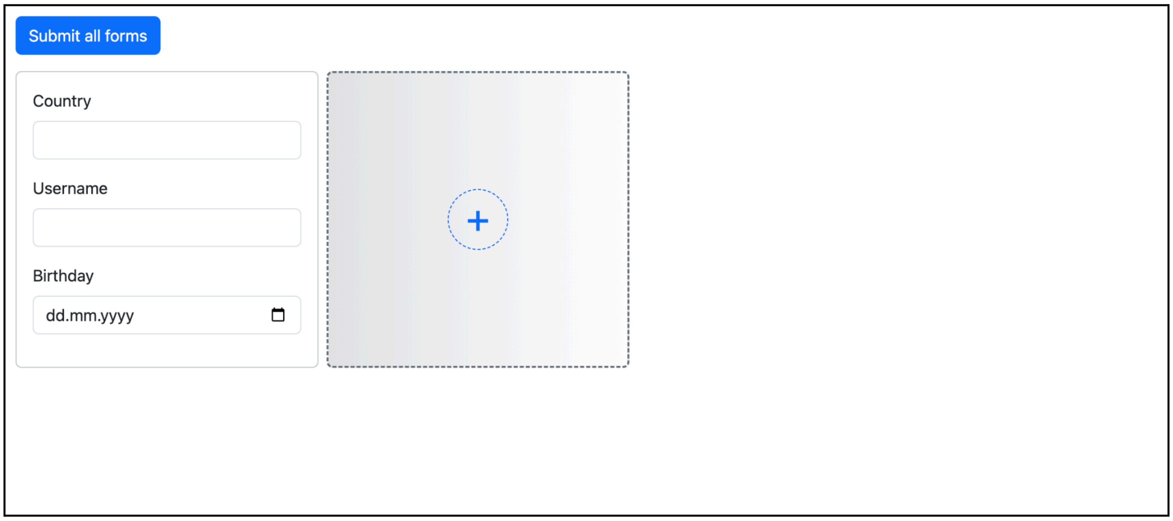
Short description:

Take the existing project from the provided archive and complete the following tasks:

1. Create an array of forms and submit them as one
2. After submitting the forms, display a timer and provide the option to cancel the submission
3. Create Directive for tooltip

Detailed description:

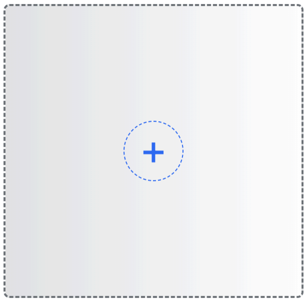
- Created a page with form cards and a Submit button. Each card should have three inputs.

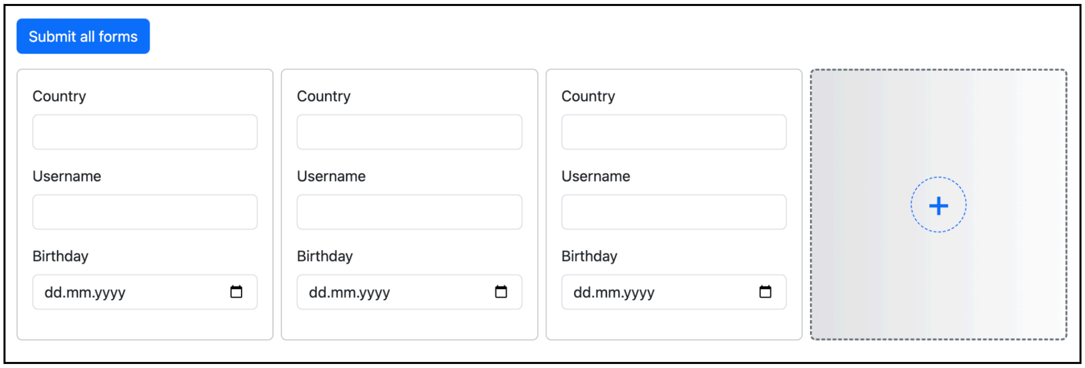


- First input, Country should be a text input. While the user types, it should suggest values from the Country enum (src/app/shared/enum/country.ts). It should validate and not allow values not listed in the Country enum.
- Second input, Username should be a text input with backend validation. After the user stops typing, the value from this input should be sent to /api/checkUsername. If the

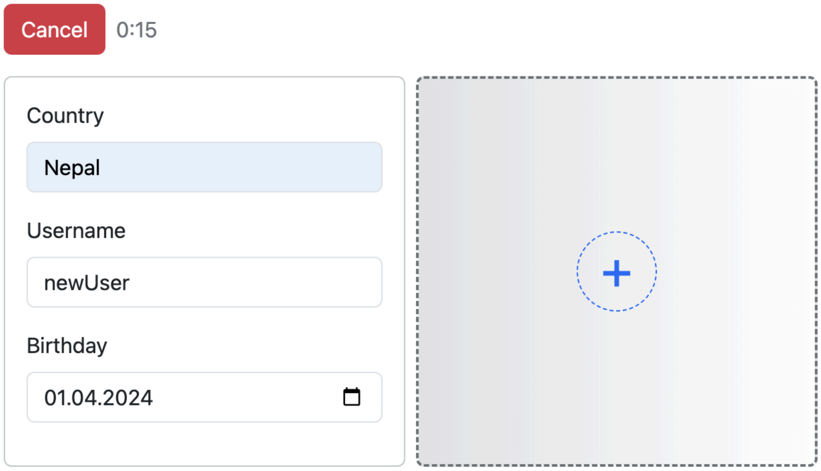
backend responds with {isAvailable: true}, it means that the username is available. If the backend responds with {isAvailable: false}, the user should choose another username. To emulate the backend, all requests are handled by an interceptor. If you need to get {isAvailable: true}, the value sent to /api/checkUsername should include the word 'new'. In other cases, /api/checkUsername returns {isAvailable: false}.

- Third input, Birthday, just datepicker. Birthdays cannot be later than the current date.
- Create a directive for input validation. If validation fails, the directive should mark the input and add text below the input: 'Please provide a correct Country/Username/Birthday' (depending on which input validation fails).
- A large button next to the form should create another form identical to the first one. Multiple forms can be created (up to 10)





- The "Submit all forms" button should block all forms from editing and trigger the timer. The "Submit all forms" button should change to a "Cancel" button. Display the timer next to the button.



Country
Nepal

Username
newUser

Birthday
01.04.2024

- If the user presses "Cancel," the timer stops, and the user can edit all forms again. If the timer runs out, the values from all forms should be sent to the /api/submitForm endpoint, the forms should be cleared, the timer should disappear, and the "Cancel" button should change back to the "Submit all forms" button.

Acceptance criteria:

- All form validations work
- Adding more form cards works
- Tooltip directive works
- Submit with timer works

Points of focus:

- Code cleanliness, following programming principles and design patterns.
- Architecture, code separation by files, project structure.

Additional Instructions:

- Markup is not important. Try to make the HTML and SCSS code as good as possible, but don't worry about the appearance of the page.
- Bootstrap is already added to the project; you are free to use it.
- You can find the backend emulator in src/app/shared/mock-backend/mock-backend.interceptor.ts.
  /api/regions - gets the country from the Country enum, returns the region.
  /api/checkUsername - gets the username, returns {isAvailable: boolean} (true if the name contains the word 'new', false in other cases).
  /api/submitForm - gets the form array value.
- If you need to explain your decisions, use comments in the code.
- If you need to provide any additional information about the project setup and usage do it in a README file.

Good luck!)