Problem statement

Title: Create an Angular web application for signing in / logging-in the user. The designs for the same can be found in the following link. Designs for Assignment

Details:

The first step is to ask the user to enter their email address or phone number.
Validate if the user already exists or not and accordingly show signup or login screen.

In login screen

1. Show a password field in addition to the email/phone number field, already entered by the user.
2. On clicking on the Login button, check if the password is valid and accordingly show a success modal or failure inline message, stating the password is not valid.
   (you can store a few pairs of email/phone numbers and password fields in mocks and validate)

In sign up screen

Step 1:

1. Show a Name field and Create Password field in addition to the email/phone number field already entered by the user.
2. On clicking on the Continue button, show the next page with the next set of fields.

Step2:

3. The next set of fields will be coming through the backend api, in our case using mocks.
   a. Organization-name - string - should be present in the list of allowed Organization-id that is present in the mocks response.
      Otherwise, show an inline error message- “Unknown organization-id”
   b. Designation - dropdown
   c. Birth-date - open a calendar
   d. City - text
   e. Pincode - verify its 6 digits and only number fields.

4. There should be a back button, on clicking on the back button from step2, the data entered on the previous page should persist.
5. On clicking on Next from step1, if some data was filled in step2, that data should also persist.

Evaluation criteria

1. The code should be modular and use strong **Typings**.
2. Interface/type should be created for mock responses and request payloads.
3. Please use Router and Stores(rxjs + ngrx).
4. A good folder structure or hierarchy is highly recommended.
5. Code should be highly readable, i.e. good naming, comments, and consistent indentations should be followed wherever possible.

Submission guidelines:

1. Please add the readme file with the details of setup instructions.
2. It would be good if just by commenting/uncommenting from a single place (data layer), all the cases can be tested.
3. Please commit the code to any of GitHub, Bitbucket, or GitLab and share the link for the same.