

Angular Test Assignment - Lazy Loading + HTTP

Background:

By default, all Angular modules are eagerly loaded and this can result in an overly large default JS bundle size on large projects, causing the site to load very slowly. Also, all defined routes are easily accessible without any restriction logic.

Problem:

Create an app with 3 arbitrary modules - 1 entry module that contains a root index page and 2 lazy loaded ones, both with a defined single, additional route. Provide a very simple UI to navigate between routes using any chosen HTML element (e.g. button).

Assume that a REST API is listening on `http://localhost:3000` url and the equivalent endpoint is `http://localhost:3000/login`. This endpoint expects following body schema:

```
login: string;
password: string;
```

To test how your code behaves, you may create a mock HTTP server, for example, using Postman.

Spec:

- As a user, I should be able to navigate to lazy loaded route when the preceding HTTP call doesn't return with error status code. On a 2xx response, the route should load.
- As the application, I should not make an API request until the user has changed the current route (don't the API until user actually tries to navigate).
- As the application, when the API has returned a successful response at least once, I shouldn't call the API again (for example, when user wants to navigate to another page).

Deliverables:

You may deliver a full working Angular CLI project, or just the most important files that solve mentioned problems.