Frontend Assignment

Angular / API

Assignment deliverables

Link to public repository in your personal GitHub account

* Make Small, Single-Purpose Commits
* Write Short, Detailed Commit Messages
* A README.md document that includes instructions to deploy your Angular project locally
* Place your README.md file in the root of your project git repository. After you have pushed all of your code (and documentation) to your GitHub repository, create [a release](https://docs.github.com/en/free-pro-team@latest/github/administering-a-repository/managing-releases-in-a-repository) on GitHub, and apply the tag version final-submission. After your release is created, you’ll find that there is now a .zip that can be downloaded from GitHub that contains a snapshot of your entire repository. Download this zip file, unpack it, and follow the instructions that you provided in your README to double-check that the evaluator will be able to run your project (this step is handy to make sure that you didn’t forget to include some key files in git). If needed, you can delete the release, make some changes, and re-release

Software & Tools

* Angular & material latest version with compatible nodeJs and npm versions
* VS code
* Firefox developer edition
* Postman
* <https://addons.mozilla.org/en-US/firefox/addon/cors-everywhere/> (If you get CORS error)
* <https://fakestoreapi.com/docs> (Get all users, Get a single user, User login APIs)

Project

1. Suppose you are hired as a Frontend developer for Atlassian.

Create a User Info Angular app with routing

* Login page (use *User login API*, suitable design with background image)
* Main menu (Home, Logout)
* User page (use *Get all users API*)
  + Material table (columns: id, email, username, password, firstname, city, actions)
  + Table Sorting (ascending and descending only)
  + Table Searching (id, email, username, firstname & city only)
  + Table paginations (5 users per page)
  + Actions column with “view” icon button (should open User details popup)
* User details popup model (use *Get a single user API*, display all details with suitable layout)
* 404 page (suitable design with image)

1. Follow <https://atlassian.design/foundations/color> , <https://atlassian.design/components/atlassian-navigation/examples> , <https://atlassian.design/content/language-and-grammar> , <https://atlassian.design/foundations/typography> Guidelines when possible (only guidelines, not code samples)
2. Use <https://atlassian.design/resources/logo-library> for logo image & <https://icon-icons.com/icon/atlassian-logo/147317> ico file for favicon
3. Follow <https://angular.io/guide/styleguide> for better application structure
4. Secure internal pages with Route Guards & display 404 page for incorrect urls
5. Keep user logged-In until he logout or browser tab closed
6. Any information stored in browser storage must be deleted on logout & browser tab close events