

#frontend-task

User management app

Description

User management app will have list of users with search and add functionalities.

Task

Your task is to build a simple user management app using [React](#) and following libraries:

- Use [redux](#) for state management
- Use [redux-thunk](#) or [redux-saga](#) for async operations
- Use one of the following libraries for form handling: [formik](#), [final-form](#) (react-final-form) or [redux-form](#)
- Use [prop-types](#) for each component
- Use [axios](#) or fetch for API communication

Requirements

- Login screen should contain simple login form (email + password). Upon successful login you should be redirected to user list screen.
- User list screen should connect to the API, get users list and display it. Users list should be in the GRID view. Each view should contain username, surname and avatar.
- User list screen should have search bar. Search bar should do automatic filtering of the users in the list. This filtering should start after at least 2 characters are entered in the search bar.
- User list screen should have an add form. Screen should have “+add user” button. Upon clicking on it, add form modal should be shown. It contains four fields: first name, last name, email, avatar URL.
- **Bonus task:** add form could have image “upload” functionality - just a dummy one, with an option to resize the image. (This image doesn't need to be uploaded anywhere, and user can have some dummy URL for the avatar)

Additional resources

- You can use the following API for data: <https://reqres.in/>
- Use any image for the default avatar URL (used for adding new users)

Deliverable

- Provide a link to a GitHub/Bitbucket repository with the completed assignment and a README.md file containing any special instructions or notes that you might find important.
- The README file should also contain a paragraph or two about some of the design decisions you made and your reasoning for making them.
- The solution should only have to be run with **npm install** and **npm start** (without global packages).

Bonus points

- Responsive design
- Naming conventions
- Input validation and error handling
- Include a test suite (good code coverage)
- Using [ESLint](#) for linting (follow best practice)
- Speed and efficiency