

## Recruitment task description

The goal is to build a simple user managing app. The app has 3 routes:

- /home
- /add
- /edit/:id

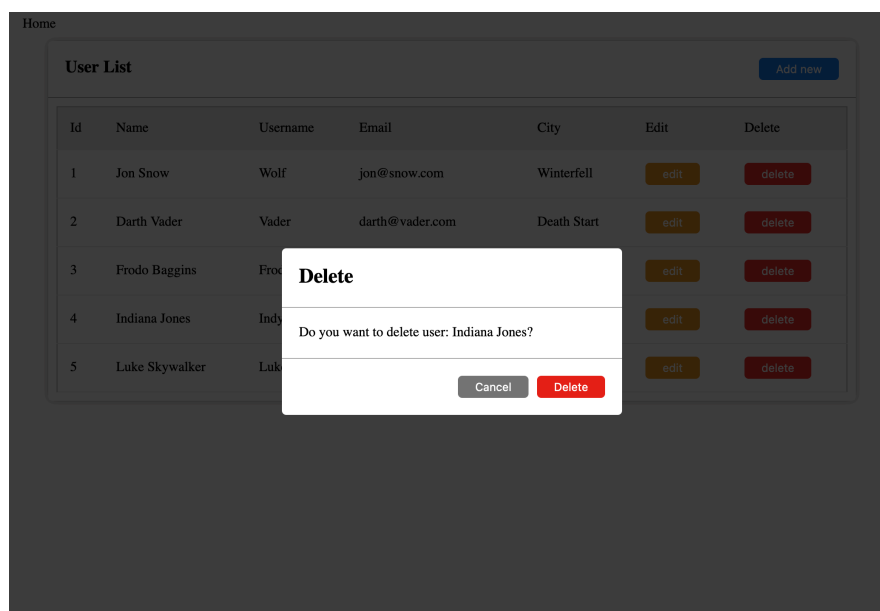
On the /home page you should display a list of users. Each user has an id, name, username, city and email. The list should have the ability to sort users by username.

Home

**User List** [Add new](#)

| Id | Name           | Username | Email              | City        | Edit                 | Delete                 |
|----|----------------|----------|--------------------|-------------|----------------------|------------------------|
| 1  | Jon Snow       | Wolf     | jon@snow.com       | Winterfell  | <a href="#">edit</a> | <a href="#">delete</a> |
| 2  | Darth Vader    | Vader    | darth@vader.com    | Death Start | <a href="#">edit</a> | <a href="#">delete</a> |
| 3  | Frodo Baggins  | Frodo    | frodo@baggins.com  | Shire       | <a href="#">edit</a> | <a href="#">delete</a> |
| 4  | Indiana Jones  | Indy     | indiana@jones.com  | Venice      | <a href="#">edit</a> | <a href="#">delete</a> |
| 5  | Luke Skywalker | Luke     | luke@skywalker.com | Tatooine    | <a href="#">edit</a> | <a href="#">delete</a> |

After clicking the ,delete' button a modal should appear. There you can delete a user or cancel the delete action. After deleting the user list should be automatically updated. Both delete and cancel buttons should hide the modal.



When you click the ,edit' button you should be redirected to the ,/edit/:id' route. There you have a form to update user data. The form should be populated with the user data you want to edit. After submitting the user list should be automatically updated. The ,cancel' button should redirect to home page.

Home

**Edit Form**

Name: Jon Snow

Username: Wolf

Email: jon@snow.com

City: Winterfell

Cancel

Submit

After clicking the ,add new' button you should be redirected to the ,/add' route. There you have a form to create a new user. After submitting the user list should be automatically updated. The ,cancel' button should redirect to /home page.

Home

**Add Form**

Name:

Username:

Email:

City:

Cancel

Submit

## Requirements

- you **must** build this app with Next.js and TypeScript
- all state management should be handled by redux-toolkit (**in this task we want to specifically test your knowledge of redux-toolkit so native react useState is NOT allowed in this app**)
- forms should be build using react-hook-form and have proper validation
- styling should be made using styled-components or emotion
- the app should have responsive styling
- for mocking api and db it is suggested to use [@mswjs/data](#) along with [@mswjs/msw](#) however you are free to use any other solution ( you can even build your own backend + db if you want ; ) )
- all success and potential error messages as well as loading states should be handled properly (be creative)
- feel free to add a couple of tests if you want
- write clean and well organized code

**After completing the task push your code to a GitHub repository and provide a link to it.**