

Middle Frontend Developer on React + 3D — Test project “Simple Dashboard 3D”

Project description

“Simple Dashboard 3D” is a React app with two pages: “Designers” and “Editor”.

On “Designers” page, it **must be** possible to view currently employed designers. There **must be** an “Add new” button, which allows the user to fill in the form and add a designer.

On “Editor” page, there is a 3D editor which **must** allow to add simple objects to a 3D canvas upon a double-click. When adding an object, a designer **must be** selected and attached to it before placing it on a canvas. Objects **must be** hoverable, selectable and movable. When being hovered or selected, an object **must** change its colour. When it is selected, it **might** be possible to edit object properties, such as attached designer, size, and colour.

Data for both designers and objects **must** come from and be saved to an API, which is not currently accessible. A mock **must be** implemented in a way that makes plugging in a real API easy. This mock **must** persist state while the app is open, and **might** persist state between app launches/refreshes.

All forms **must** have proper validation rules.

This project might take up to 4-5 hours to complete. UX choices are up to you.

Resulting project **must** be shared as source code of a runnable package, and **might** be deployed somewhere (e.g. GitHub Pages).

Model definitions

Designer:

- Full name
- Working hours
- Attached objects count

Object:

- Name
- Attached designer
- Color
- Position
- Size (small, normal, large)

Acceptance criteria

Non-functional criteria:

- It is implemented using reasonably new versions of React and other libraries
- It uses a separate routing library
- It uses a separate state management library
- Reasonable tests are a bonus
- In the repository, there are instructions on how to run the project locally
- If it is also deployed, the deployed version works the same as shared version
- Addressing accessibility issues is a bonus

Functional criteria:

- “Must”s are implemented, “might”s are optional