

Forms

Basics, design, and UI

Day 1 - Fundamentals of Frontend Development

Demo

1

2

3

username

student_1001234

password

.....

Previous

Next

1

2

3

Street Address

Postal code

Please select your state

▼

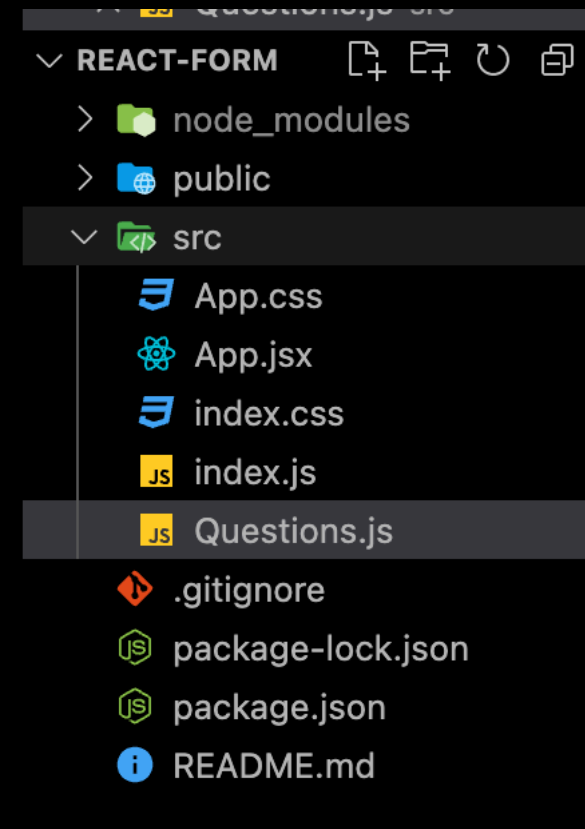
Previous

Next

Setup

`npx create-react-app react-form-bootstrap`

```
export const questions = [
  {
    section: 1,
    items: [
      {
        label: "username",
        type: "text",
        value: "username",
      },
      {
        label: "password",
        type: "password",
        value: "password",
      },
    ],
  },
  {
    section: 2,
    items: [
      {
        label: "Street Address",
        type: "text",
        value: "street",
      },
      {
        label: "Postal code",
        type: "text",
        value: "postalcode",
      },
      {
        label: "State",
        type: "select",
        value: "state",
        options: ["State 1", "State 2"],
      },
    ],
  },
  {
    section: 3,
    items: [
      {
        label: "If you are ready to submit please press `Submit`",
        type: "information",
      },
    ],
  },
];
```



prepare `Questions.js`

Progress bar



Form Item

text type

A diagram of a form item. At the top is a horizontal progress bar with three circular markers labeled 1, 2, and 3. Marker 1 is on the left, marker 2 is in the middle, and marker 3 is on the right. Below the progress bar is a solid blue rectangular button with the text "Next" centered on it. Below the button is a text label "label-of-this-form-item". Below the label is a light gray rectangular input field with a thin blue border. The input field contains the text "test".

Form Item

password type

1

2

3

Next

label-of-this-form-item

.....|

Form Item

select type

1

2

3

Next


Please select your state

✓ State 1

State 2

Multi-page Form

No validation



1 2 3

Street Address

Postal code

Please select your state

▼

Previous

Next

Multi-page Form

Back button preserves answer

1

2

3

Street Address

8 somapah road

Postal code

123456

State 1

▼

Previous

Next

Validation with yup

npm i yup

```
const USERREGEX = /^[A-Za-z0-9_]+$/;
const POSTALCODEREGEX = /^[0-9]{6}$/;

const usernameAndPassword = object().shape({
  username: string()
    .required("Username is required")
    .min(8, "Username must be at least 8 characters long")
    .matches(
      USERREGEX,
      "Username can only contain alphanumeric characters and underscores"
    ),
  password: string()
    .required("Password is required")
    .min(8, "Password must be 8 characters long")
    .matches(/\d/, "Password requires a number")
    .matches(/[a-z]/, "Password requires a lowercase letter")
    .matches(/[A-Z]/, "Password requires an uppercase letter")
    .matches(/[^\w]/, "Password requires a symbol"),
});
```

Validation with yup

npm i yup

```
usernameAndPassword
  .validate(pagesAnswers[index], { abortEarly: false })
  .then((responseData) => {
    console.log("no validation errors");
    console.log(responseData);
    goToNextPage();
  })
  .catch((err) => {
    console.log(err);
    console.log(err.name); // ValidationError
    console.log(err.errors);
    // set the error message
    setValidated(false);
    setErrorMessage(err.errors.join("\r\n"));
  });
```

Summary

- **Validate** content
- **Persistent** (relevant) data when back is pressed
- **Progress** indicator
- **Labels** and **Keys**
- **Verbose** error message
- **Confirmation**