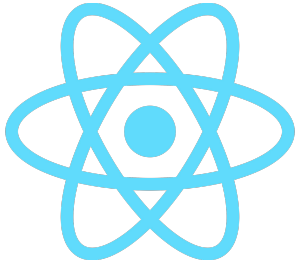


React.js

Lecture 5

Agenda

- Recap last lecture points
- Forms
- Native vs third party forms handler
- General Topics
- Open discussion and questions

Forms

Forms in React

We can handle forms in different ways:

Here's the most native and simple ways to handle it using hooks,

https://www.freecodecamp.org/news/how-to-build-forms-in-react/

With React 19 new hooks appeared to handle the forms on servers directly (self study)

server-side functions you can call directly from your React components (or form submissions) without building a separate API layer.

https://www.youtube.com/watch?v=ExZUdkfu-KE&t=443s

Forms in React

For More Organized, complex and advanced forms handling, You can use forms third parties like

- Formik with Yup for validation

  https://formik.org/

- React hook form with Zod

  https://react-hook-form.com/

Extra Topics

Code Splitting

Bundling is great, but as your app grows, your bundle will grow too. Especially if you are including large third-party libraries. You need to keep an eye on the code you are including in your bundle so that you don’t accidentally make it so large that your app takes a long time to load.

To avoid winding up with a large bundle, it’s good to get ahead of the problem and start “splitting” your bundle

https://dev.to/franklin030601/code-splitting-in-react-js-4o2g

Code Splitting

import()

The best way to introduce code-splitting into your app is through the dynamic `import()` syntax.

React.lazy

The `React.lazy` function lets you render a dynamic import as a regular component.

Code Splitting

Before:

```
import OtherComponent from './OtherComponent';
```

After:

```
const OtherComponent = React.lazy(() => import('./OtherComponent'));
```

This will automatically load the bundle containing the `OtherComponent` when this component is first rendered.`React.lazy` takes a function that must call a dynamic `import()`. This must return a `Promise` which resolves to a module with a `default` export containing a React component.

Code Splitting

The lazy component should then be rendered inside a Suspense component, which allows us to show some fallback content (such as a loading indicator) while we're waiting for the lazy component to load.

Example:

```
import { Suspense } from 'react';

<Suspense fallback={<div>Loading...</div>}>
    <OtherComponent />
</Suspense>
```

Deployment

https://create-react-app.dev/docs/production-build

npm run build

creates a `build` directory with a production build of your app. Inside the `build/static` directory will be your JavaScript and CSS files. Each filename inside of `build/static` will contain a unique hash of the file contents. This hash in the file name enables long term caching techniques.

Env variables

https://vite.dev/guide/env-and-mode

- Create .env file at the same level of src folder
- Any environment variable should start with VITE then the key name like
    - VITE_APP_BASE_URL=http://test.com
- You can use the env variable in you code using the following
    - {import.meta.env.VITE_APP_BASE_URL}

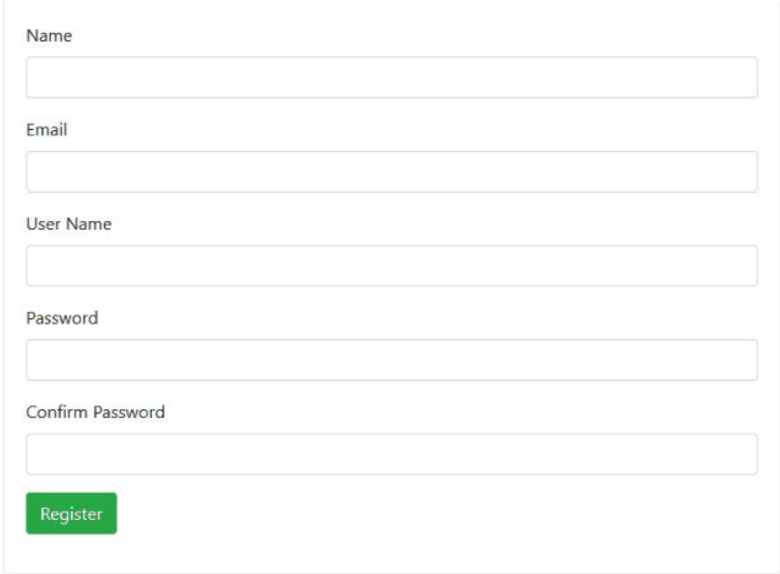
Thank you

Lap

Task :Register form

In the register page with the following fields. [Native]

- Email address [required—email format] (regex)
- Name [required]
- Username [required—contains no spaces]
- Password [required—password length not less than 8 characters, contains at least one lowercase letter, one uppercase letter, at least one digit, and a special character [example: *@%$#] (regex)
- **Example of a valid password: P@ssword1234**
- Confirm password: [required—matches previous password]
- [Bonus] Try to show all errors when the user clicks on the register button at the beginning.
- After the user submits the form, please alert the object of data and redirect the user to the home page (products list)



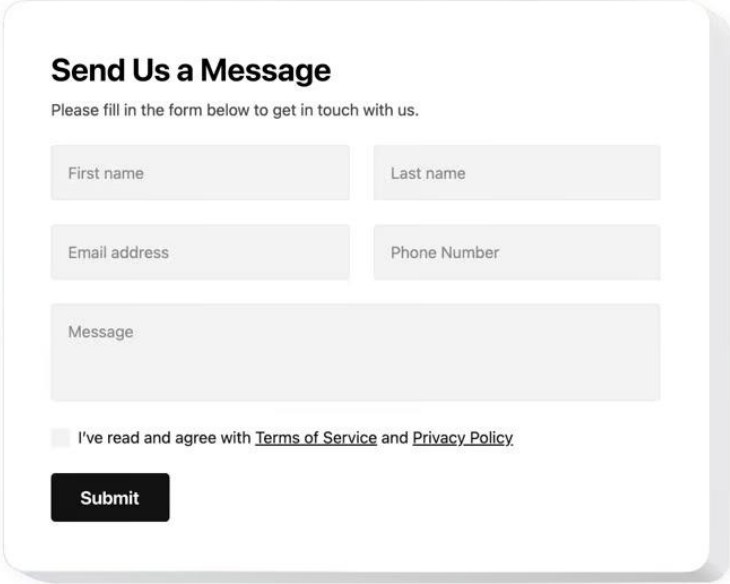
Task :Contact us form

In the contact us page apply the following fields with validation [Using third party]

- Email address [required—email format]
- First Name [required]
- Last Name [required]
- Phone Number [optional]
- Message [Required, min 10 characters , max 500]

After submit , show a success message that we will get to you soon, then reset the form values.

- [Bonus] Extra, try to use a phone number package with countries and phone number formats



Ecommerce App

General enhancements for the tasl

- Apply code splitting for pages.
- Add api base url to env file