Skyforge Coding

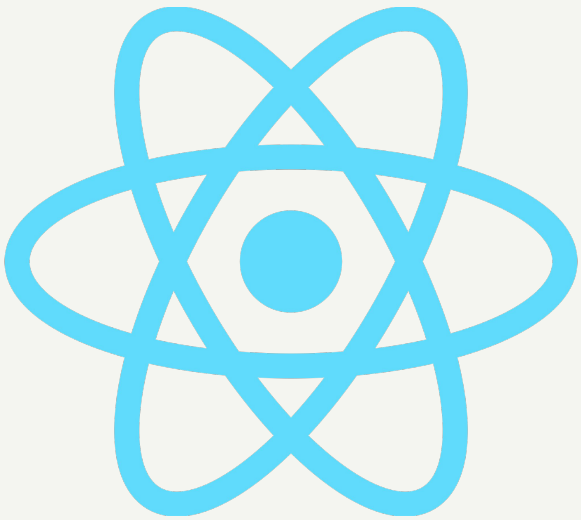
Conquer libraries in React.js projects

Eric Santana

Improving React.js

A Comprehensive Guide to Essential Libraries.

This ebook is your roadmap to becoming proficient in React.js development, with a focus on leveraging key libraries such as Vite, Immer, React Hook Form, Material UI, and React Router. Each chapter will provide practical examples and insights to help you understand how these libraries enhance your React.js projects.



Skyforge Coding - Eric Santana

Chapter 1

Accelerate Development with Vite

Accelerate Development with Vite

a next-generation build tool

Vite is an innovative build tool, to streamline your React.js development process.
This tool leverages native ES module imports to offer a fast development experience.

Accelerate Development with Vite

Vite eliminates the need for bundling during development, it serves your code directly from an optimized development server.
Let's get started and look how to **forge** it in your project.

```
npm init @vitejs/app my-react-app --template react
cd my-react-app
npm install
npm run dev
```

Capítulo 2

Tame the state management with Immer

Tame the state with Immer

Simplify State Management

With the power of Immer, object mutation no longer needs to be made throughout arrays.

```
const [state, setState] = React.useState([])
setState(produce(draft ⇒ {
    draft.push({ text: 'forge with Immer' })
}))
```

Immer allows you to work with immutable objects in a way that feels mutable. thus removing the need to create new objects when updating a state.

Tame the state with Immer

Comparison

For the comparison of the old way and the immer way to renew the state, let's say we want to revise the quantity of iron in stock:

Component.jsx

```
const [rawMetals, setRawMetals] =
    React.useState({
        iron: {
            Quantity: 10
        }
    })
```

Tame the state with Immer

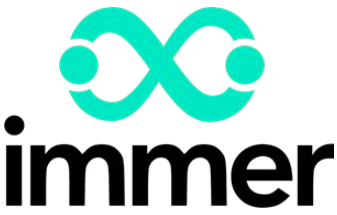
Comparison

Without Immer:

```
const updatedRawMetalsWithoutImmer = {
    ...rawMetals,
    iron: {
        ...rawMetals.iron,
        Quantity: rawMetals.iron.Quantity + 5
    }
};
setRawMetals(updatedRawMetalsWithoutImmer);
```

With Immer:

```
const updatedRawMetalsWithImmer =
produce(rawMetals, draft ⇒ {
    draft.iron.Quantity += 5;
});
setRawMetals(updatedRawMetalsWithImmer);
```



Capítulo 3

Unlock the Pathways with React Router

Unlock the Pathways with React Router

Effortless Navigation

React Router allows you to navigate between different parts of your app without actually reloading the page.

It simplifies routing in React.js applications, allowing intuitive and dynamic navigation between different components.

React Router

Unlock the Pathways with React Router

Effortless Navigation

Some advantages of using React Router in your project:

- Single-page Application (SPA) Support.
- You can define your routes in a declarative way using JSX.
- Nested Routing: you to create complex page structures with multiple levels.
- Dynamic Routing: You can dynamically render routes

Unlock the Pathways with React Router

Effortless Navigation

Behold a example showcasing the implementation of React Router in your code.

Navigation.jsx

```jsx
import React from 'react';
import {
    BrowserRouter as Router,
    Route,
    Link }
from 'react-router-dom';

const Home = () => <h1>Home</h1>;
const About = () => <h1>About</h1>;

const App = () => {
  return (
    <Router>
      <nav>
        <Link to="/">Home</Link>
        <Link to="/about">About</Link>
      </nav>
      <Route path="/" exact component={Home} />
      <Route path="/about" component={About} />
    </Router>
  );
};
```

Capítulo 4

Streamline Forms with React Hook Form

Streamline Forms with React Hook Form

Handling Forms Better

At its core, React Hook Form is a library that leverages the power of React's hooks to streamline form handling.

Wield the power of hooks to effortlessly capture form data, validate inputs, and handle form submission with finesse.

Streamline Forms with React Hook Form

Handling Forms Better

Let's initialize the React hook form with the resources inside his useForm hook:

```
import React from 'react'
import { useForm } from 'react-hook-form';

function MyForm() {
  const {
    register,
    handleSubmit,
    formState: { errors },
    setValue,
    watch
  } = useForm();
  ...
}
```

The returned values of this hook will be use for basic use, validation and controlled components.

Streamline Forms with React Hook Form

Basic Usage

Basic Form with Input Field:

```
function MyForm() {
...
  const onSubmit = data => {
    console.log(data);
  };
  return (
    <form onSubmit={handleSubmit(onSubmit)}>
      <input
        {...register("firstName")}
        placeholder="First Name" />
      <input
        {...register("lastName")}
        placeholder="Last Name" />
      <button type="submit">Submit</button>
    </form>
  );
}

export default MyForm;
```

Streamline Forms with React Hook Form

Validation

Form with Validation:

```
function MyForm() {
...
  const onSubmit = data => {
    console.log(data);
  };
  return (
    <form onSubmit={handleSubmit(onSubmit)}>
      <input
      {...register("email", { required: true })}
      placeholder="Email" />
      {errors.email &&
      <span>Email is required</span>}

      <input
      {...register("password", {
            required: true, minLength: 6
          })}
      type="password" placeholder="Password" />
      {errors.password &&
      <span>Password must be at least 6
      characters long</span>}

      <button type="submit">Submit</button>
    </form>
  );
}

export default MyForm;
```

Streamline Forms with React Hook Form

Controlled

Form with Controlled Components:

```
function MyForm() {
...
  const onSubmit = data ⇒ {console.log(data);};
  return (
    <form onSubmit={handleSubmit(onSubmit)}>
      <input
      {...register("email")}
      placeholder="Email" />

      <input
      {...register("password")}
      type="password"
      placeholder="Password" />

      <input {...register("confirmPassword")}
      type="password" placeholder="Confirm Password"
        onChange={e ⇒ {
          setValue("confirmPassword", e.target.value);
        }}
      />
      {password !== watch("confirmPassword")
      && <span>Passwords do not match</span>}

      <button type="submit">Submit</button>
    </form>
  );
}
export default MyForm;
```

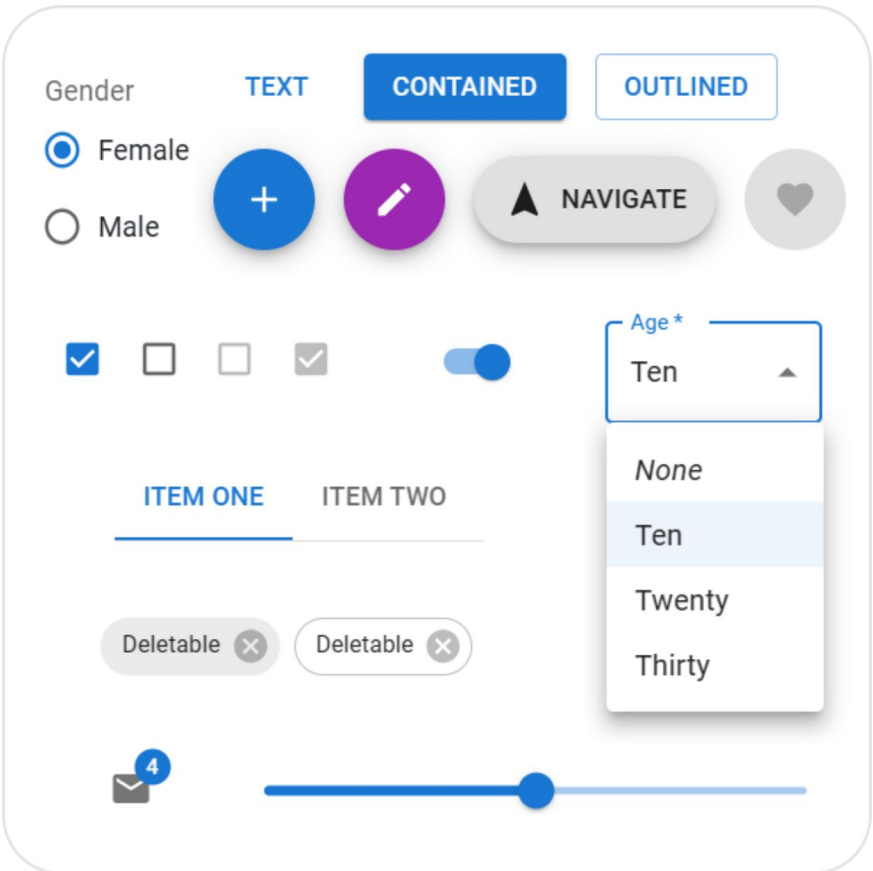
Capítulo 5

Crafting Excellence: Material-UI for React

Crafting Excellence: Material-UI for React

Enhance UI

Material-UI simplifies the UI development process by providing ready-to-use components that follow Material Design guidelines. This ensures consistency and enhances user experience across applications.



Crafting Excellence: Material-UI for React

Enhance UI

Let's dive into a code example to illustrate how to use Material-UI components in a React application.

```
import React from 'react';
import {
    Button,
    Typography
    } from '@mui/material';

function App() {
    return (
        <div>
            <Typography variant="h1">
                    Welcome to Material-UI
            </Typography>
            <Button variant="contained"color="primary">
                    Get Started
            </Button>
        </div>
    );
}
export default App;
```

Crafting Excellence: Material-UI for React

Theming

If you want to change the look and feel of a component, you can customize it in multiple ways. In the code below the Overriding Styles:

```
import React from 'react';
import { Button } from '@mui/material';

function CustomButton() {
    return (
        <Button
        sx={{ backgroundColor: 'red',
        borderRadius: '10px' }}>
            Click Me
        </Button>
    );
}

export default CustomButton;
```

Crafting Excellence: Material-UI for React

Enhance UI

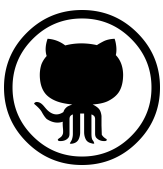
Remember, mastering Material-UI takes time and practice. Start by experimenting with small projects and gradually incorporate more advanced features as you become more comfortable. With dedication and perseverance, you'll soon become proficient in leveraging Material-UI to build stunning user interfaces.

Ending

Thank you for reading this far.

This book was produced by a human and in collaboration with an AI, with several sections generated by artificial intelligence and edited and typeset by a human.
This book was produced for educational purposes and free reading, do not resell.

Connect with me



https://github.com/efms25

https://www.linkedin.com/in/eric-santana-955812159/