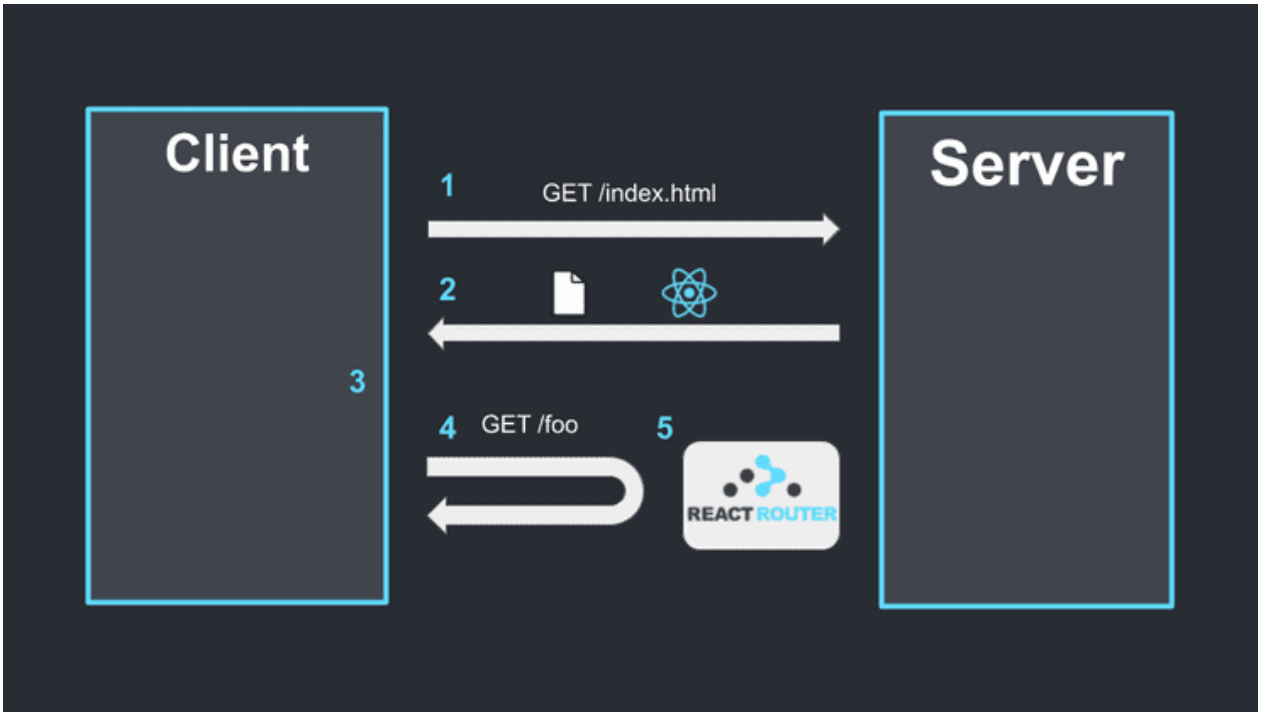
Analysis Of Folder Structure



React Router is a standard library for routing in React. It enables the navigation among views of various components in a React Application, allows changing the browser URL, and keeps the UI in sync with the URL.

Let us create a simple application to React to understand how the React Router works. The application will contain three components: home component, about a component, and contact component. We will use React Router to navigate between these components.

- Create React App doesn't include page routing.
- React Router is the most popular solution.

Add React Router

```
npm i -D react-router-dom
```

Folder Structure

To create an application with multiple page routes, let's first start with the file structure.

Within the src folder, we'll create a folder named pages with several files:

src\pages\:

- Home.js
- Blogs.js
- Contact.js

Each file will contain a very basic React component:

Home.js:

```
const Home = () => {
  return <h1>Home</h1>;
};

export default Home;
```

Blogs.js:

```
const Blogs = () => {
  return <h1>Blog Articles</h1>;
};

export default Blogs;
```

Contact.js:

```
const Contact = () => {
  return <h1>Contact Me</h1>;
};

export default Contact;
```

Using React Router

Now we will use our Router in our index.js file.

Example:

Use React Router to route to pages based on URL:

Index. js:

```
import ReactDOM from "react-dom";
import { BrowserRouter as Router, Switch, Route, Link } from
"react-router-dom";
import Home from "./pages/Home";
import Blogs from "./pages/Blogs";
import Contact from "./pages/Contact";

export default function App() {
  return (
    <Router>
      <div>
        <Link to="/">Home</Link>
      </div>
      <div>
        <Link to="/blogs">Blog Articles</Link>
      </div>
```

```
      <div>
        <Link to="/contact">Contact Me</Link>
      </div>
      <hr />
      <Switch>
        <Route exact path="/">
          <Home />
        </Route>
        <Route path="/blogs">
          <Blogs />
        </Route>
        <Route path="/contact">
          <Contact />
        </Route>
      </Switch>
    </Router>
  );
}

ReactDOM.render(<App />, document.getElementById("root"));
```

Example Explanation:

- We wrap our content first with <Router>.
- <Link> is used to set the URL and keep track of browsing history.
- Anytime we link to an internal path, we will use <Link> instead of <a href="">.
- <Switch> is similar to a JavaScript switch statement. It will conditionally render the <Route> that matches the <Link> path.