

Sure! Here's a **Simple Structure for a 5-page ReactJS Website** that includes a header, footer, and menu with CSS Flexbox.

The project will consist of the following components:

1. **App.js** - The main entry point for the app.
2. **Header.js** - The navigation menu.
3. **Footer.js** - The footer.
4. **Home.js** - The home page.
5. **About.js** - The about page.
6. **Services.js** - The services page.
7. **Contact.js** - The contact page.
8. **NotFound.js** - A 404 not found page.
9. **App.css** - The main CSS file with Flexbox layout.

## Step 1: Setting up the project

First, let's create a new React app. If you haven't already set up React, you can create a new project with the following commands:

cmd code:-

```
npx create-react-app react-website
cd react-website
npm start
```

## Step 2: Install React Router

Since we want to create a multi-page website, we'll use `react-router-dom` to handle routing.

cmd code:-

```
npm install react-router-dom
```

## Step 3: Create the components

Let's start by creating the necessary components in the `src` folder.

### 1. App.js

jsx code:-

```
import React from 'react';
import { BrowserRouter as Router, Route, Switch } from 'react-router-dom';
import Header from './Header';
import Footer from './Footer';
import Home from './Home';
import About from './About';
import Services from './Services';
import Contact from './Contact';
```

```

import NotFound from './NotFound';
import './App.css';

function App() {
  return (
    <Router>
      <div className="app-container">
        <Header />
        <main>
          <Switch>
            <Route path="/" exact component={Home} />
            <Route path="/about" component={About} />
            <Route path="/services" component={Services} />
            <Route path="/contact" component={Contact} />
            <Route component={NotFound} />
          </Switch>
        </main>
        <Footer />
      </div>
    </Router>
  );
}

export default App;

```

## 2. Header.js

```

jsx
Copy code
import React from 'react';
import { Link } from 'react-router-dom';

function Header() {
  return (
    <header className="header">
      <div className="logo">My Website</div>
      <nav className="nav">
        <ul>
          <li><Link to="/">Home</Link></li>
          <li><Link to="/about">About</Link></li>
          <li><Link to="/services">Services</Link></li>
          <li><Link to="/contact">Contact</Link></li>
        </ul>
      </nav>
    </header>
  );
}

export default Header;

```

## 3. Footer.js

```

Jsx code
import React from 'react';

```

```
function Footer() {
  return (
    <footer className="footer">
      <p>&copy; 2024 My Website. All rights reserved.</p>
    </footer>
  );
}

export default Footer;
```

## 4. Home.js

### Jsx code

```
import React from 'react';

function Home() {
  return (
    <div className="page-content">
      <h1>Welcome to the Home Page</h1>
      <p>This is the home page of the website.</p>
    </div>
  );
}

export default Home;
```

## 5. About.js

### jsx code

```
import React from 'react';

function About() {
  return (
    <div className="page-content">
      <h1>About Us</h1>
      <p>Learn more about us here.</p>
    </div>
  );
}

export default About;
```

## 6. Services.js

### jsx code

```
import React from 'react';

function Services() {
  return (
    <div className="page-content">
      <h1>Our Services</h1>
      <p>We offer a wide range of services.</p>
    </div>
  );
}
```

```
export default Services;
```

## 7. Contact.js

jsx code

```
import React from 'react';

function Contact() {
  return (
    <div className="page-content">
      <h1>Contact Us</h1>
      <p>Get in touch with us here.</p>
    </div>
  );
}

export default Contact;
```

## 8. NotFound.js

jsx code

```
import React from 'react';

function NotFound() {
  return (
    <div className="page-content">
      <h1>404 - Page Not Found</h1>
    </div>
  );
}

export default NotFound;
```

## Step 4: Add the CSS (App.css)

Now, let's style the components using Flexbox.

css code:-

```
/* App.css */

/* General Styles */
body {
  font-family: Arial, sans-serif;
  margin: 0;
  padding: 0;
}

.app-container {
  display: flex;
  flex-direction: column;
  min-height: 100vh;
```

```
}

/* Header Styles */
.header {
  background-color: #333;
  color: white;
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 10px 20px;
}

.header .logo {
  font-size: 1.5rem;
}

.nav ul {
  display: flex;
  list-style-type: none;
  margin: 0;
  padding: 0;
}

.nav ul li {
  margin: 0 15px;
}

.nav ul li a {
  color: white;
  text-decoration: none;
  font-size: 1rem;
  padding: 5px 10px;
}

.nav ul li a:hover {
  background-color: #444;
  border-radius: 5px;
}

/* Main content Styles */
main {
  flex: 1;
  padding: 20px;
  background-color: #f4f4f4;
}

.page-content {
  background-color: white;
  padding: 20px;
  border-radius: 5px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

/* Footer Styles */
.footer {
  background-color: #333;
  color: white;
}
```

```
text-align: center;
padding: 10px;
}
```

## Step 5: Running the App

Now you should be able to run your app with the following command:

```
cmd (terminal):-
```

```
npm start
```

## File Structure Overview:

```
/src
|-- /components
|   |-- Header.js
|   |-- Footer.js
|   |-- Home.js
|   |-- About.js
|   |-- Services.js
|   |-- Contact.js
|   |-- NotFound.js
|-- App.js
|-- App.css
|-- index.js
```

## Final Notes:

- **Routing:** React Router (`react-router-dom`) handles navigation between pages. We use `<Route />` to map paths to components.
- **Flexbox Layout:** The header and footer are laid out using Flexbox, and the content area (main) takes up the available space.
- **Responsiveness:** You can expand the CSS further to ensure the layout is responsive, adding media queries if needed.