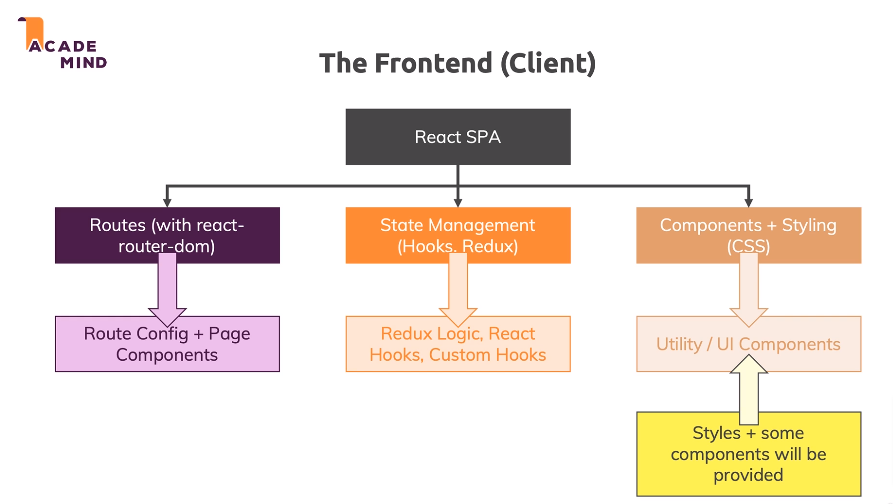
**The Frontend (Client Side)**

****

**React SPA:** React is in charge of re-rendering everything in the browser. Only one HTML Page that single page that makes up this terms single page application is surfed from server to the browser in the end, and thereafter React takes over. And whenever something needs to be drawn onto the screen, or whenever something needs to changes on the screen or needs to be re-rendered, react will do that.

**Routes (with react-router-dom):**For that we’ll also handle frontend or *client side routing* with an extra library “*react-router-dom*” and this wil helpus render totally different react components. React is all about component.

React router will help us reander React compnents basd on a path the user enters into the URL bar of the browser, so we will have the feeling of having multiple pages, but in the end it’s all react and therefore JavaScript re-rendering significatnt parts of our pages so of what the user seen when we change the URL or when we get new data from the back end and so on.

**State Management (Hooks, Redux):**Now in our React application, we also typically handle some front end state.

State in the end is data, that influences what is shown on the screen. And when our state changes, chances are that we want to re-render parts of the screen.

React is all about managing such state and it has built in mechanisms to automatically re-render the parts of the screen that need to change when something changes.

Now for state management, we can use react state in class based components.

We can use hooks relatively kind of new feature in React and or also application wide state the Redux Library.