

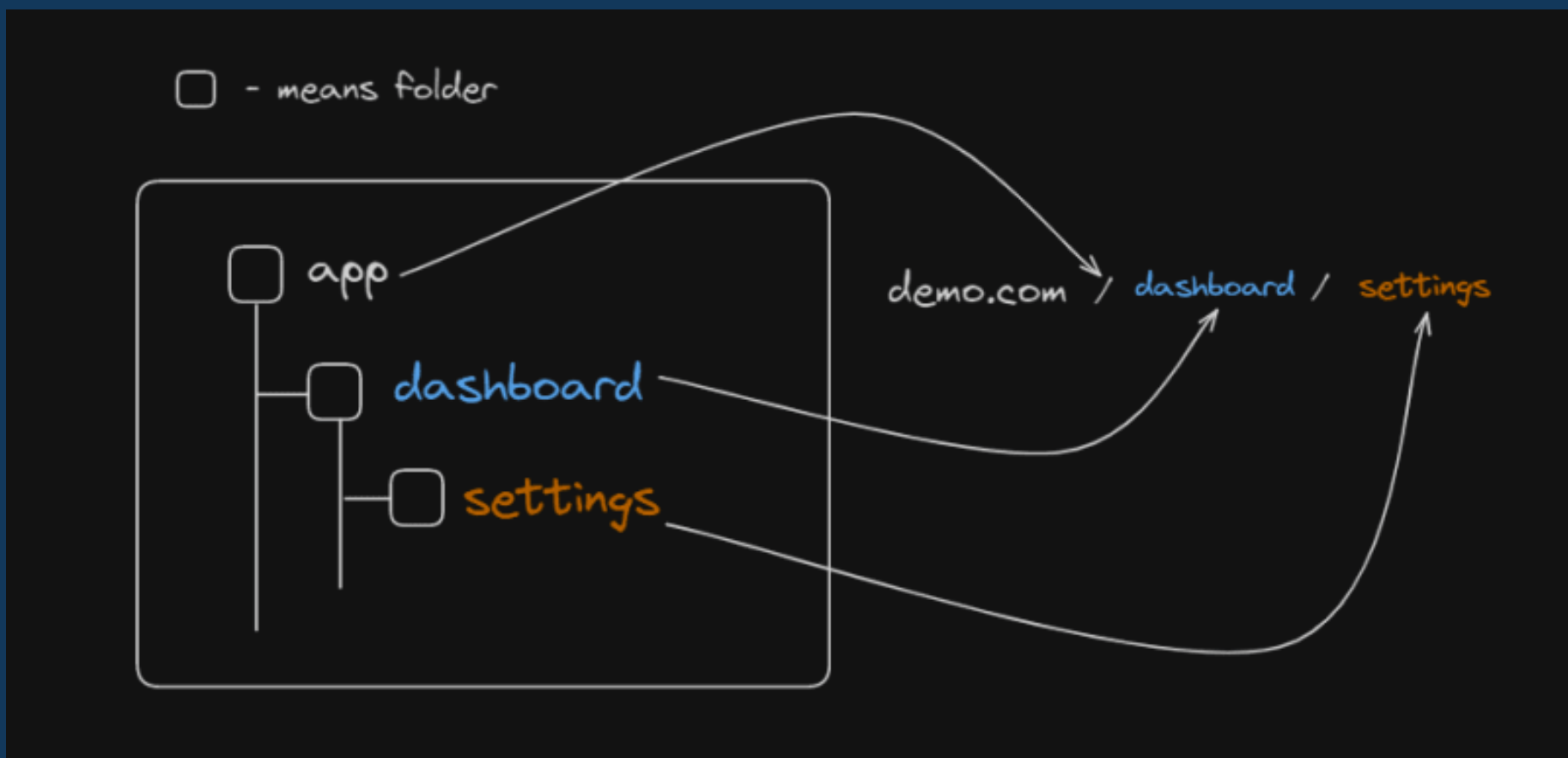


Navigating Your Next.js App: Mastering Routing

This post dives into routing concepts in Next.js, focusing on the latest app-based routing structure:

SWIPE

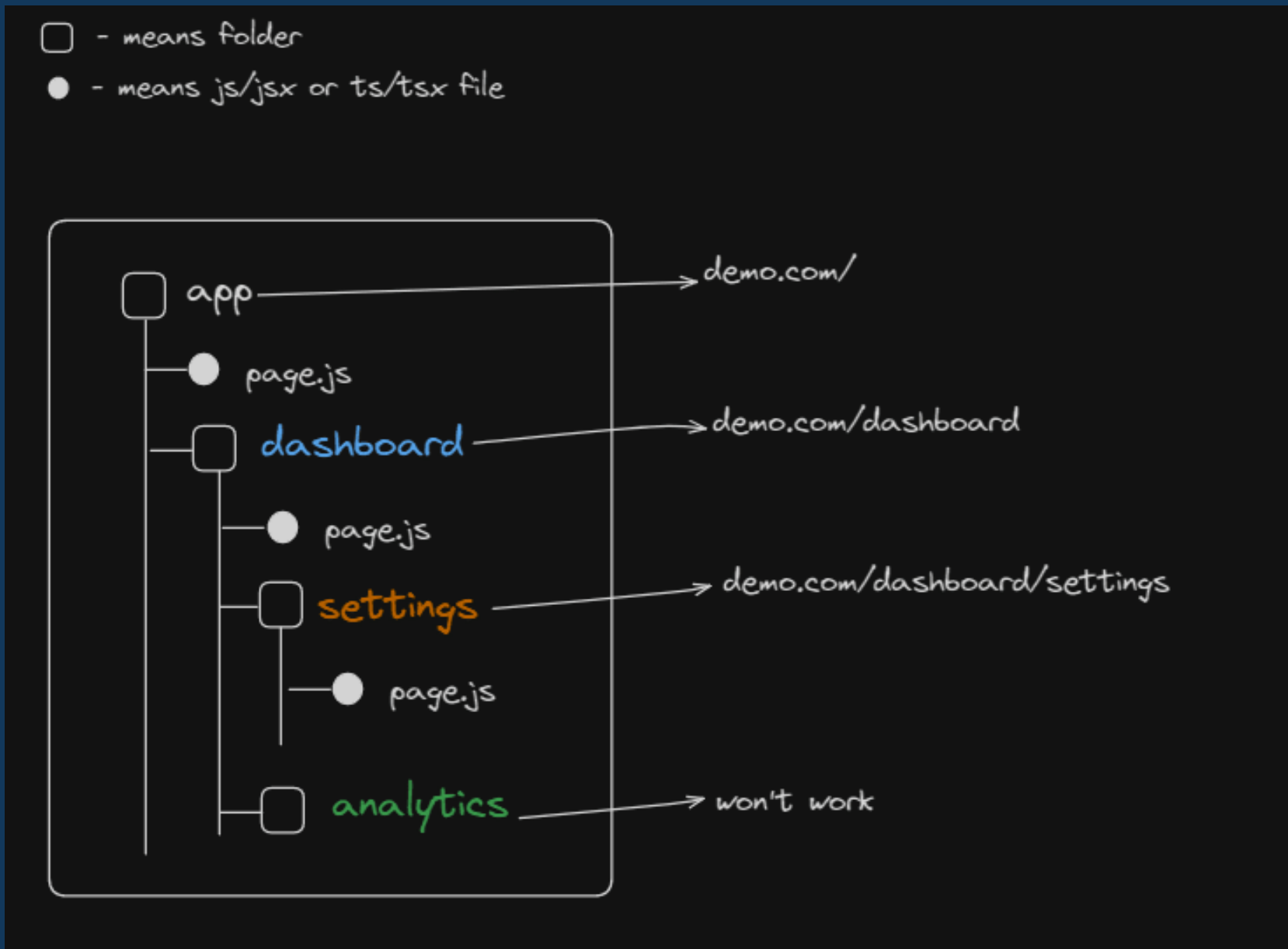




Understanding Routing in Next.js

This introductory slide emphasizes the importance of routing in Next.js applications, similar to a map guiding users through different pages and sections.

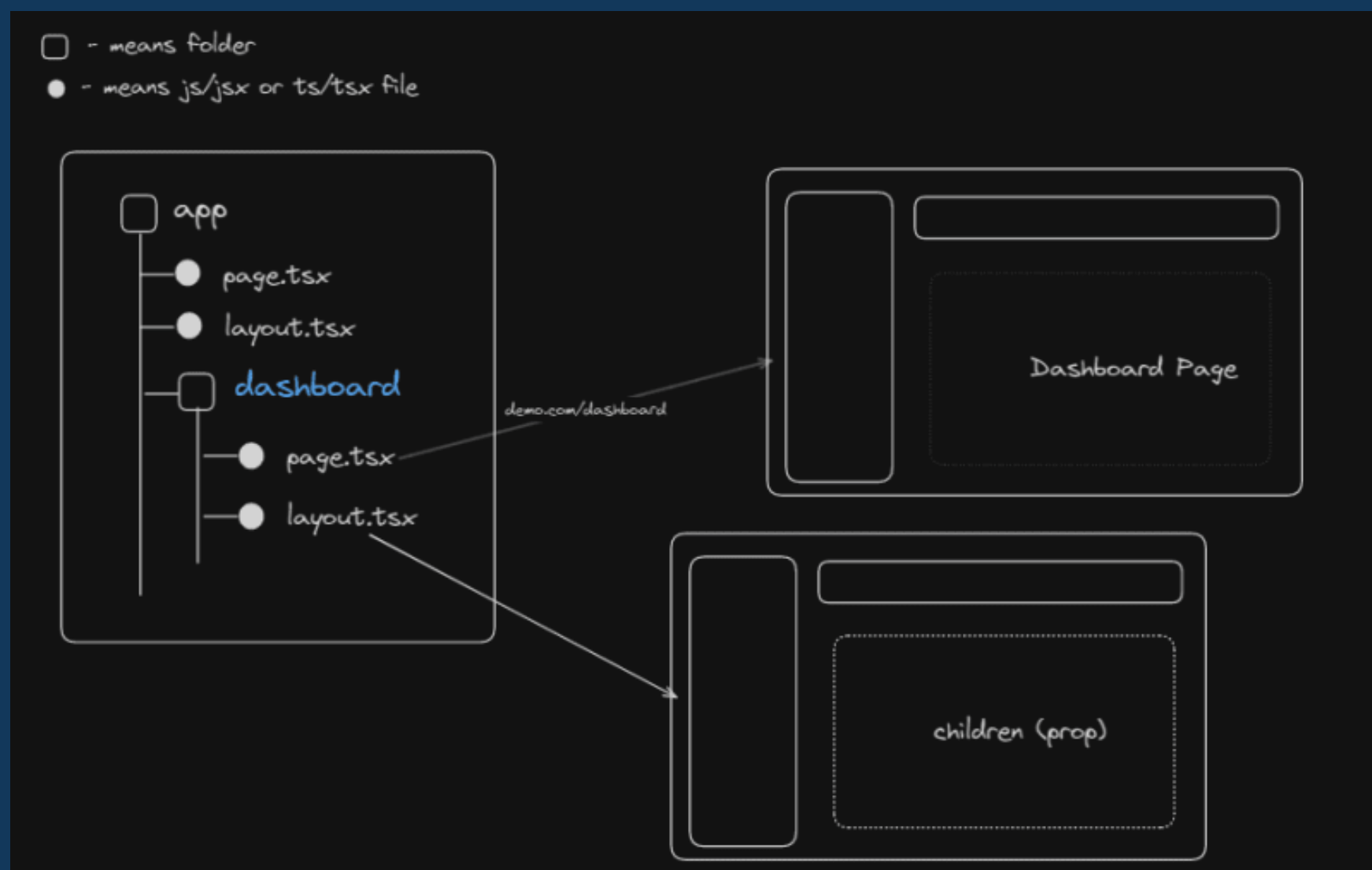
KEEP SWIPING



The Power of Pages:

we can use a special file page.js to make our pages accessible. If you don't add any page.js file inside route folder we won't see anything on the route.





The Power of Layouts:

A layout.tsx file is UI that is shared between multiple pages. This layout serves as the template for your other sub-pages of the same group.

E.g in above img the children prop in the layout.tsx allows you to render pages or components within the dashboard structure.

ONE MORE

Route	Example URL	params
<code>app/blog/[slug]/page.js</code>	<code>/blog/a</code>	<code>{ slug: 'a' }</code>
<code>app/blog/[slug]/page.js</code>	<code>/blog/b</code>	<code>{ slug: 'b' }</code>
<code>app/blog/[slug]/page.js</code>	<code>/blog/c</code>	<code>{ slug: 'c' }</code>

Dynamic Routing: Flexibility at its Finest

Next.js supports dynamic routing using square brackets (`[slug].js`). This allows for routes that can vary based on user input or data, like product pages with unique IDs.

COMMENT BELOW