

Section 1: Getting Started

Sunday, September 08, 2024 11:25 AM

What is NextJS? Why would you use it?





NextJS is a fullstack React framework

It vastly **simplifies** the process of building fullstack applications with React



Trend: Build fullstack apps instead of SPAs

Get the **best of both worlds**:

Highly interactive frontends, blending seamlessly with **tightly connected backends**



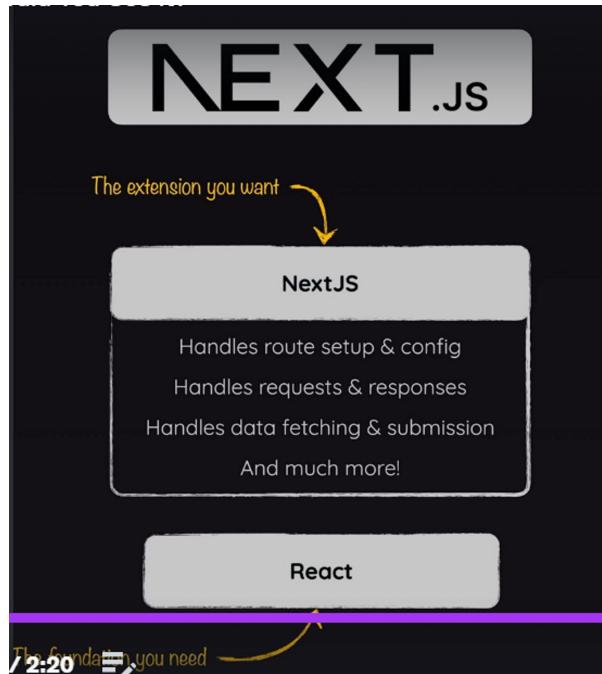
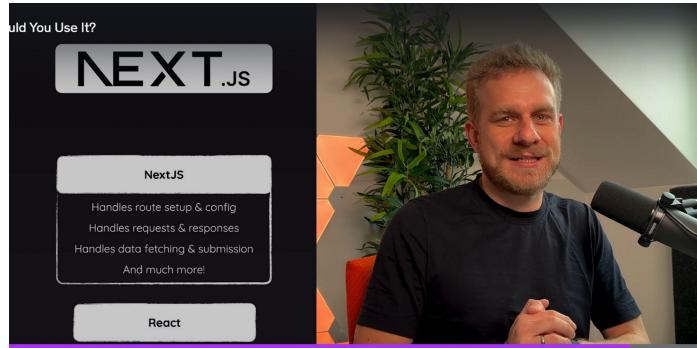
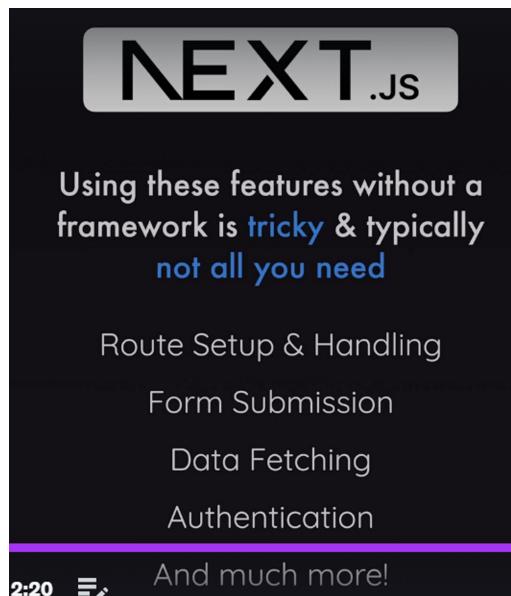
React is becoming a hybrid library

Improved server-side rendering

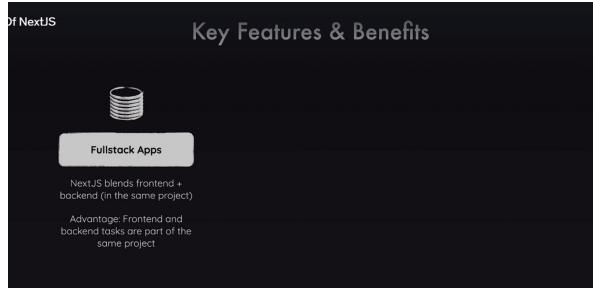
Streaming responses

React Server Components

Server Actions



Key Features and Benefits of NextJS



```

12
13  async function Meals() {
14    console.log('Fetching meals');
15    const meals = await getMeals();
16
17    return <MealsGrid meals={meals} />;
18  }

```

Fetching and Rendering data can be easier & more efficient.

```

const db = sql('meals.db');

export async function getMeals() {
  // await new Promise((resolve) => setTimeout(resolve, 5000));
  // throw new Error('Loading meals failed');
  return db.prepare('SELECT * FROM meals').all();
}

```

Handling Form Submission can be easier and more secure.

```

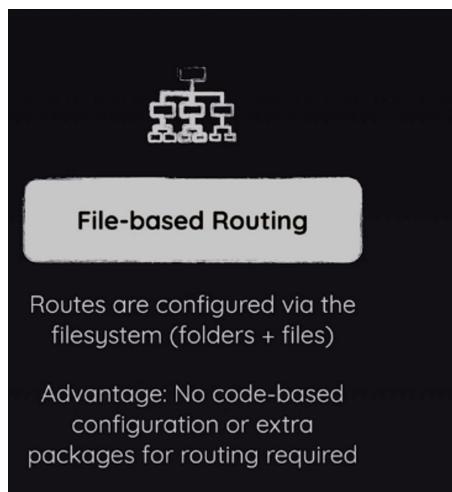
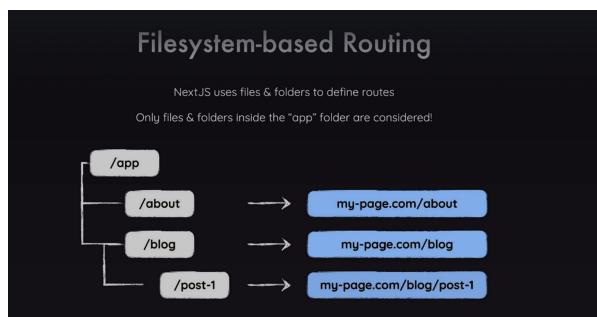
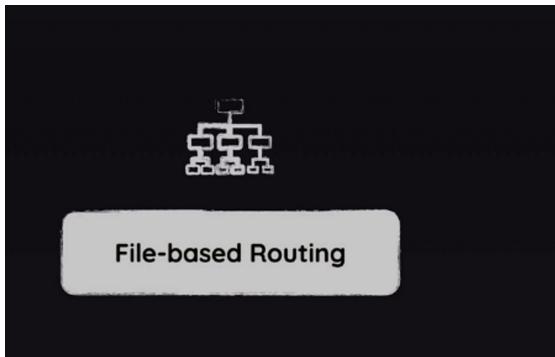
import MealsFormSubmit from '@/components/meals/meals-form-submit';

export default function ShareMealPage() {
  const [state, formAction] = useFormState(shareMeal, { message: null });
}

return (
  <>
    <header className={classes.header}>
      <h1>
        | Share your <span className={classes.highlight}>favorite meal</span>
      </h1>
      <p>Or any other meal you feel needs sharing!</p>
    </header>
    <main className={classes.main}>
      <form className={classes.form} action={formAction}>
        <div className={classes.row}>
          <label htmlFor="name">Your name:</label>
          <input type="text" id="name" name="name" required />
        </div>
        <label htmlFor="email">Your email:</label>
        <input type="email" id="email" name="email" required />
      </form>
    </main>
  </>
)

```

Another big advantage is file-based routing.



p

Creating a First NextJS App

```
npx create-next-app@latest
```

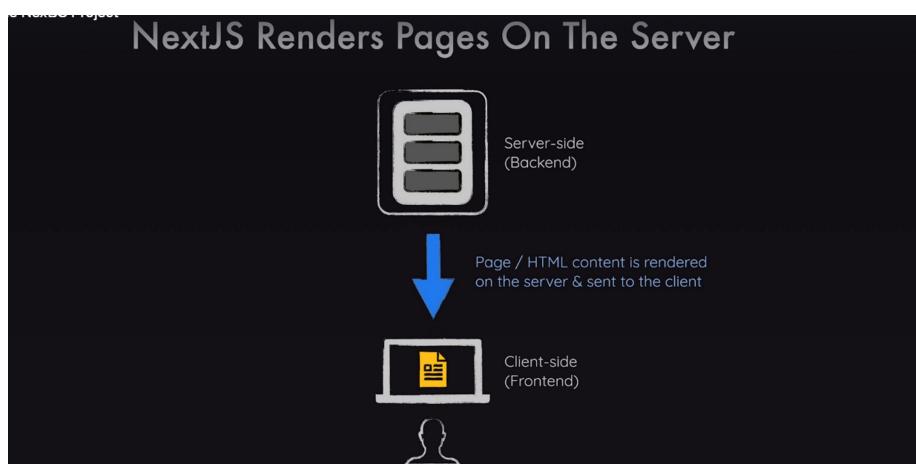
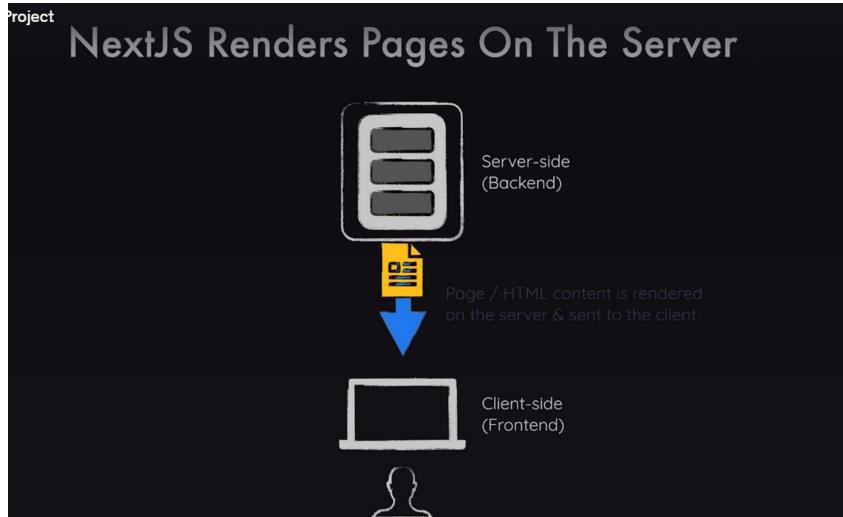
NextJS vs "Just React" - Analyzing The NextJS Project

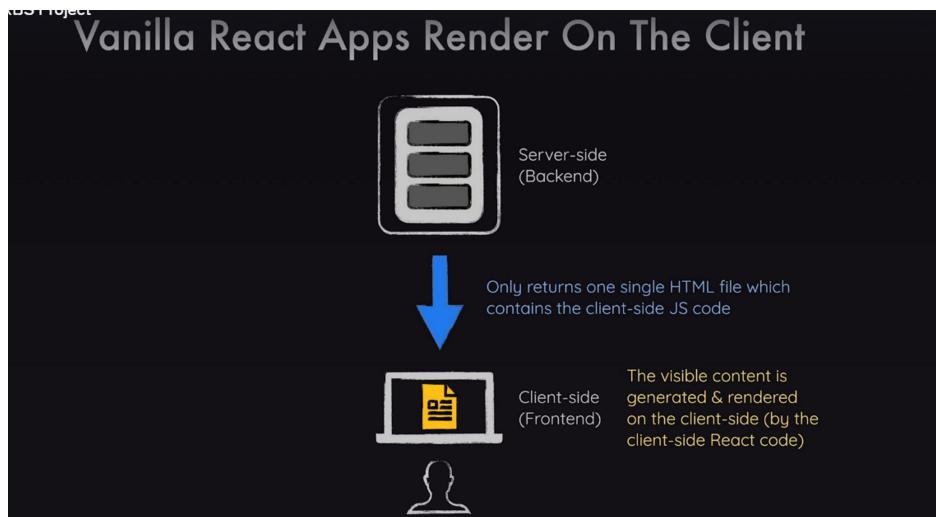
You can see the code under the div tag,

```

internal.js" async=""></script><script src="/_next/static/chunks/main-100.js"></script><script src="/_next/static/chunks/main-34199.js"></script><script src="/_next/static/chunks/polyfills.js" type="image/png" sizes="500x486"></script><script src="/_next/static/chunks/main.js" module="main"><img alt="A server surrounded by magic sparks" data-bbox="11 26 100 100"/><h1>Welcome to this NextJS Course!</h1><p>Let's get started!</p><main><script src="/_next/static/chunks/webpack.js?v=170126068338" async=""></script><script src="/_next/static/chunks/app-pages-internals.js" module="app-pages-internals"><h2>Next.js push API</h2><pre>next_f.push([1,2,3])</pre><script src="/_next/static/chunks/app-pages-internals.js" module="app-pages-internals"><h2>Next.js push API</h2><pre>next_f.push([1,2,3])</pre></script></main></body></html>

```





```

3  <head>
4    <script type="module">
5      import RefreshRuntime from "/@react-refresh"
6      RefreshRuntime.injectIntoGlobalHook(window)
7      window.$RefreshReg$ = () => {}
8      window.$RefreshSig$ = () => (type) => type
9      window._vite_plugin_react_preamble_installed_ = true
10     </script>
11
12    <script type="module" src="@vite/client"></script>
13
14    <meta charset="UTF-8" />
15    <link rel="icon" type="image/svg+xml" href="/vite.svg" />
16    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
17    <title>Vite + React</title>
18  </head>
19  <body>
20    <div id="root"></div>
21    <script type="module" src="/src/main.jsx?t=1701364942024"></script>
22  </body>
23</html>
24

```



React JS is by default a client side Javascript library, which means it runs in the browser and it edits and manipulates the page in the browser.



This UI update is performed via client-side
React code - the page is NOT reloaded
or re-rendered on the server!

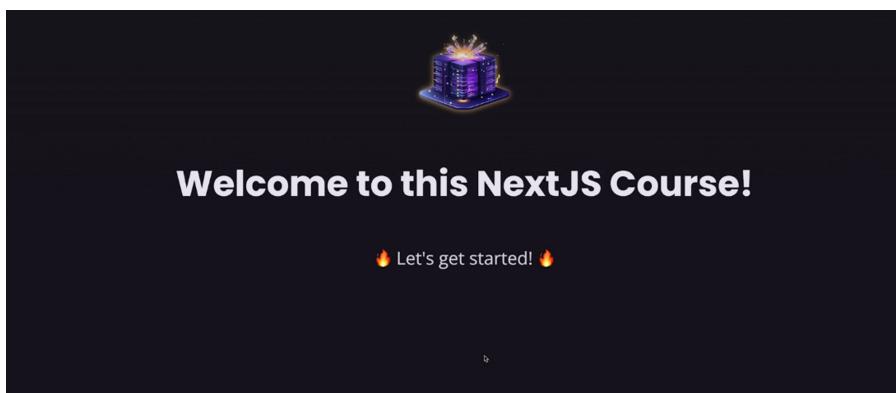
count is 4 *

Edit src/App.jsx and save to test HMR

And that's amazing! Because the page can be edited after it's been loaded. We can update the state of that page, after it's been loaded without having to fetch a new page.

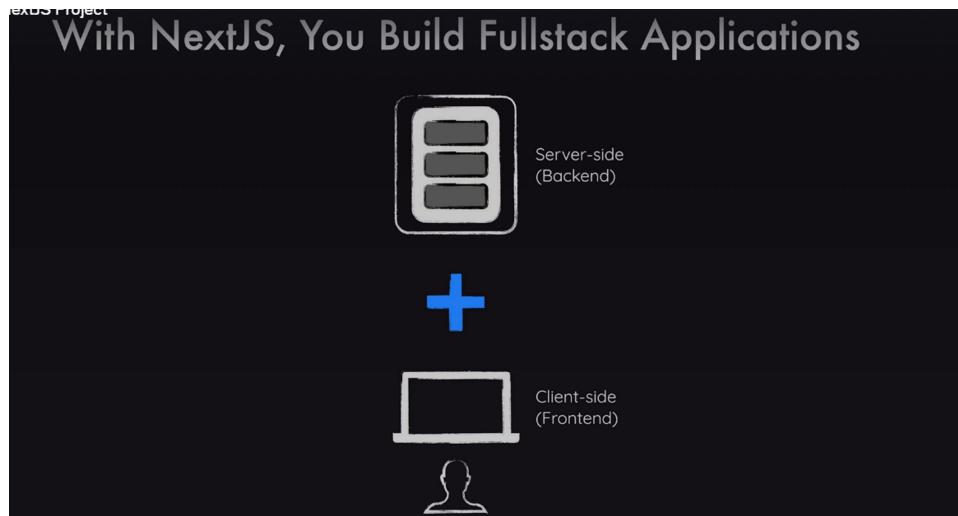
But the disadvantage is, search engine crawlers don't see the actual page content, if they visit the page for the first time.

And that's the difference here with Next JS. Here we got all the content, not just visually on the screen, but in the actual source code.



```
internals.js" async=""></script><title>NextJS Course App</title><meta name="description" content="Your first NextJS app!"><link rel="icon" href="/icon.png" type="image/png" sizes="500x486"/><script src="/_next/chunks/polyfills.js" nomodule=""></script><head></head><body><div>Welcome to this NextJS Course!</div><div>Let's get started! <span>Edit src/App.jsx and save to test HMR</span></div><div>count is 4 *</div><div>Edit src/App.jsx and save to test HMR</div></body></html>
```

With Next JS, you no longer have a client-side application, instead you have a full stack application.



With NextJs Projects, you have code that runs on the client and you have a lot of code that runs on the server. It's both.

Pages in Next.Js are simply React Components.

Page Router vs App Router - One Framework, Two Approaches

```
$ npx create-next-app@latest
✓ What is your project named? ... first-app
✓ Would you like to use TypeScript? ... No / Yes
✓ Would you like to use ESLint? ... No / Yes
✓ Would you like to use Tailwind CSS? ... No / Yes
✓ Would you like to use `src/` directory? ... No / Yes
? Would you like to use App Router? (recommended) > No / Yes
```

Two Approaches For Building NextJS Apps



Pages Router



App Router

Two Approaches For Building NextJS Apps



Pages Router

Has been around for many years

Very stable

Used in many existing NextJS projects

Allows you to build feature-rich fullstack apps with React



App Router

Introduced with NextJS 13

Marked as stable but still relatively new & partially buggy

Supports modern Next & React features (→ fullstack React apps)

The future of NextJS