

## Here's a step-by-step guide to building a 5–6 page website with Next.js

### ✓ Step 1: Create the Next.js App:-

```
npx create-next-app my-website
```

```
cd my-website
```

```
C:\Users\User>F:
F:\>cd Raj singh
F:\Raj Singh>npx create-next-app my-website
Need to install the following packages:
create-next-app@15.3.4
Ok to proceed? (y) y

√ 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 your code inside a `src/` directory? ... No / Yes
√ Would you like to use App Router? (recommended) ... No / Yes
√ Would you like to use Turbopack for `next dev`? ... No / Yes
√ Would you like to customize the import alias (`@/*` by default)? ... No / Yes
Creating a new Next.js app in F:\Raj Singh\my-website.

Using npm.

Initializing project with template: app

Installing dependencies:
- react
- react-dom
- next
```

Note : Important points while creating Project Folder

√ Would you like to use TypeScript? ... No

√ Would you like to use ESLint? ... No

√ Would you like to use Tailwind CSS? ... No

√ Would you like your code inside a `src/` directory? ... Yes

√ Would you like to use App Router? (recommended) ... Yes

√ Would you like to use Turbopack for `next dev`? . Yes

√ Would you like to customize the import alias (`@/\*` by default)? ... No

Note:- Prompt In Chat Gpt-

in my project folder there is src and inside src there is app folder inside app folder there is two files page.js and layout.js

You're using the **new App Router** (`/app` directory) introduced in **Next.js 13+**, instead of the traditional `pages` router.

That changes the structure — but it's even more powerful! Let's walk through it step-by-step.

---

## ✓ App Router Basics (You're Using This)

Your project structure looks like:

```
src/
├── app/
│   ├── page.js           → Home page (`/`)
│   └── layout.js        → Shared layout (nav, footer, etc.)
```

Next.js now uses **file-based routing** via **folders**, and each folder represents a route.

---

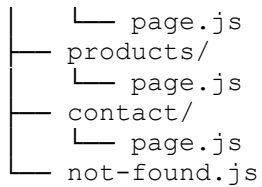
## ✓ Step-by-Step: Add 5–6 Pages Using App Router

### Step-by-Step: Add 5–6 Pages Using App Router

## 1. 📁 Folder Structure

Create folders for each route inside `app/`:

```
src/
├── app/
│   ├── layout.js
│   ├── page.js           → Home
│   ├── about/
│   │   └── page.js
│   └── services/
```



## ✓ App Router Basics (You're Using This)

Your project structure looks like:

```
src/
├── app/
│   ├── page.js           → Home page (`/`)
│   └── layout.js        → Shared layout (nav, footer, etc.)
```

src/app/globals.css file code :-

```
/* styles/globals.css */

nav {
  background: #222;
  padding: 10px 20px;
  display: flex;
  justify-content: space-between;
  align-items: center;
  color: white;
}

nav .logo {
  font-weight: bold;
  font-size: 18px;
}

nav .menu a {
  color: white;
  margin-right: 15px;
  text-decoration: none;
}

nav .menu a:last-child {
  margin-right: 0;
}

main {
```

```
padding: 20px;
}

footer {
  background: #eee;
  padding: 10px;
  text-align: center;
}
```

## 2. layout.js — Shared Layout

```
// app/layout.js or app/layout.tsx

import './globals.css'; // adjust the path if needed
import Link from 'next/link';

export default function RootLayout({ children }) {
  return (
    <html lang="en">
      <body>
        <nav>
          <div className="logo">MyLogo</div>
          <div className="menu">
            <Link href="/">Home</Link>
            <Link href="/about">About</Link>
            <Link href="/services">Services</Link>
            <Link href="/products">Products</Link>
            <Link href="/contact">Contact</Link>
          </div>
        </nav>

        <main>{children}</main>

        <footer>
          © 2025 MyWebsite
        </footer>
      </body>
    </html>
  );
}
```

### 3. 🏠 page.js — Home Page

```
// src/app/page.js
export default function HomePage() {
  return (
    <div>
      <h1>Welcome to My Website</h1>
      <p>This is the homepage using App Router.</p>
    </div>
  );
}
```

### 4. About Page (about, services, products, contact)

Each page needs a `page.js` file. Here's an example:

```
// src/app/about/page.js
export default function AboutPage() {
  return (
    <div>
      <h1>About Us</h1>
      <p>We are passionate about building modern web apps.</p>
    </div>
  );
}
```

### 5. ✕ Optional: not-found.js (Custom 404 Page)

```
// src/app/not-found.js
export default function NotFound() {
  return (
    <div>
      <h1>404 - Not Found</h1>
      <p>Sorry, the page you are looking for doesn't exist.</p>
    </div>
  );
}
```

✓ services/page.js

```
// src/app/services/page.js
export default function ServicesPage() {
  return (
    <div>
      <h1>Our Services</h1>
      <ul>
        <li>Web Development</li>
        <li>Mobile App Development</li>
        <li>UI/UX Design</li>
        <li>SEO Optimization</li>
      </ul>
    </div>
  );
}
```

✓ products/page.js

```
// src/app/products/page.js
export default function ProductsPage() {
  const products = [
    { id: 1, name: 'Laptop', price: '₹49,999' },
    { id: 2, name: 'Smartphone', price: '₹19,999' },
    { id: 3, name: 'Smartwatch', price: '₹7,499' },
  ];

  return (
    <div>
      <h1>Our Products</h1>
      <div style={{ display: 'flex', gap: '20px', flexWrap: 'wrap' }}>
        {products.map(product => (
          <div
            key={product.id}
            style={{
              border: '1px solid #ccc',
              padding: '15px',
              borderRadius: '8px',
              width: '200px'
            }}
          >
            <h3>{product.name}</h3>
            <p>Price: {product.price}</p>
            <button>Buy Now</button>
          </div>
        ))}
      </div>
    </div>
  );
}
```

✓ contact/page.js

```
// ✓ src/app/contact/page.js
export default function ContactPage() {
  return (
    <div>
      <h1>Contact Us</h1>
      <form style={{ maxWidth: '400px' }}>
        <div style={{ marginBottom: '15px' }}>
          <label>Name</label><br />
          <input type="text" placeholder="Your name" style={{ width: '100%' }}
required />
        </div>
        <div style={{ marginBottom: '15px' }}>
          <label>Email</label><br />
          <input type="email" placeholder="Your email" style={{ width: '100%'
}} required />
        </div>
        <div style={{ marginBottom: '15px' }}>
          <label>Message</label><br />
          <textarea rows="4" style={{ width: '100%' }} placeholder="Your
message" required />
        </div>
        <button type="submit">Send Message</button>
      </form>
    </div>
  );
}
```

✓ All these pages will be accessible from:

- /services
- /products
- /contact

And they're automatically included in your shared `layout.js`

## Start your local server with:

Open Terminal & Navigate to Project Folder

```
cd path/to/your/project
```

For example:

```
cd my-website
```

```
npm run dev
```

You'll see something like:

```
Local: http://localhost:3000
```

Output:-

# Welcome to My Website

This is the homepage using App Router.