

Introduction to Next.js

A full-stack framework for React

by William Khine

DevOps Team Lead @ PACS

terrahacks.ca/next.pdf

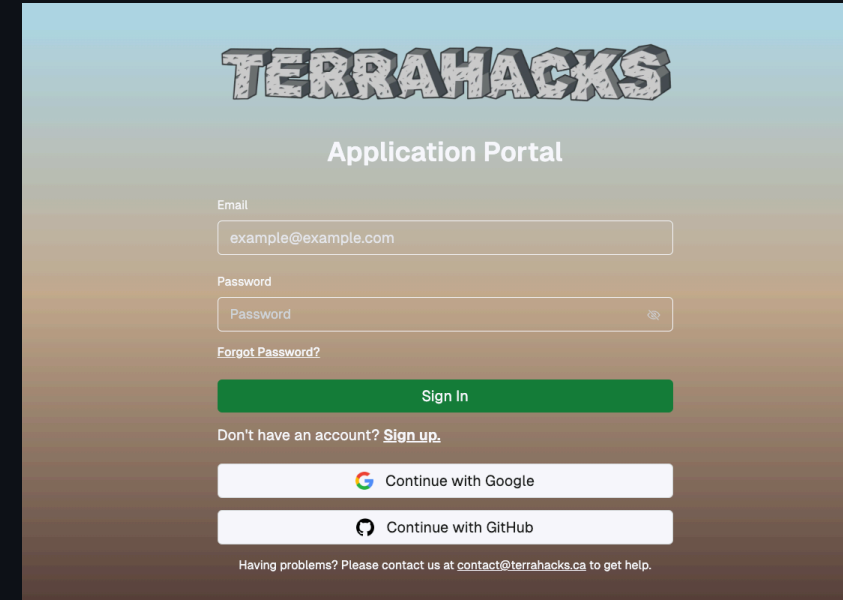


About me

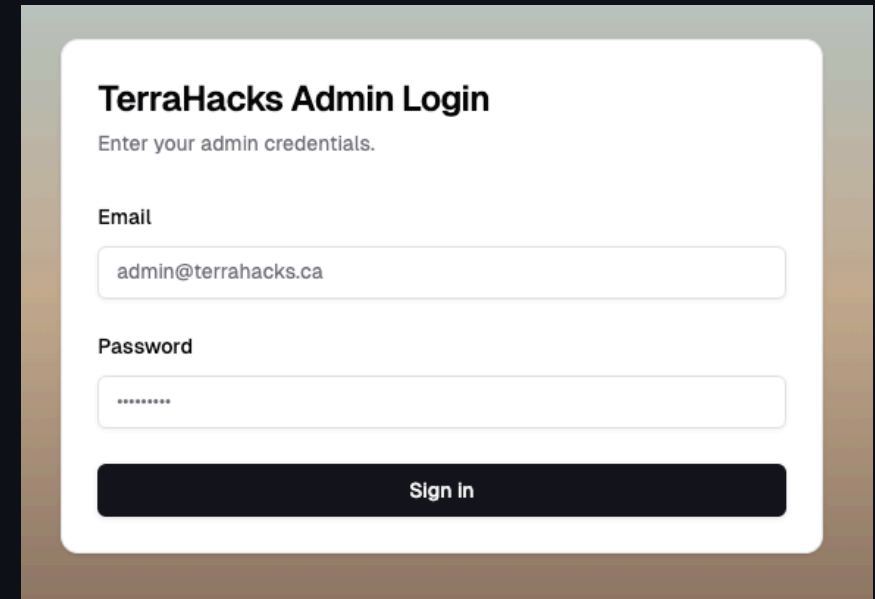
- Computer Science @ TMU
- DevOps Team Lead @ PACS
- 5+ Full-stack projects in Next.js
- GitHub: [@williamKhine](#)
- [khine.dev](#)

Project Highlights

- TerraHacks Portal
- TerraHacks Admin Portal
- USSTM Website



The screenshot shows the 'TERRAHACKS' logo at the top in a stylized, blocky font. Below it is the title 'Application Portal'. The login form includes an 'Email' field with the placeholder 'example@example.com', a 'Password' field with a placeholder 'Password' and a toggle icon, and a 'Forgot Password?' link. A green 'Sign In' button is positioned below the password field. Below the button is a link: 'Don't have an account? [Sign up.](#)'. Two social login buttons are present: 'Continue with Google' (with the Google logo) and 'Continue with GitHub' (with the GitHub logo). At the bottom, a small text line reads: 'Having problems? Please contact us at contact@terrahacks.ca to get help.'



The screenshot shows the 'TerraHacks Admin Login' page. The title 'TerraHacks Admin Login' is at the top, followed by the instruction 'Enter your admin credentials.'. The form contains an 'Email' field with the placeholder 'admin@terrahacks.ca' and a 'Password' field with a placeholder of eight asterisks. A dark blue 'Sign In' button is located at the bottom of the form.

What you'll be able to do after

- Start a Next.js project
- Understand Next.js project structure
- Understand components, pages, and layouts
- Understand routing in Next.js
- Understand Dynamic Routes
- Perform data fetching and Server Actions

Agenda

1. What is Next.js?
2. Why Next.js?
3. Getting Started
4. Project Structure (Routing, Components, Layouts)
5. Next.js built-in components (Image, Link)
6. Client Components
7. Server Components
8. Props
9. Dynamic Routes
10. Data Fetching

What is Next.js?

What is Next.js?

Next.js is a full-stack framework for React. Its features include server-side rendering, static site generation, and Image Optimization.

Developed by Vercel, it's a popular choice for building web applications.

Why Next.js?

Why Next.js?

- **SEO Friendly:** Server-side rendering and static site generation
- **Performance:** Automatic code splitting, optimized images, and lazy loading

Notable Websites built with Next.js

TerraHacks Portal

TERRAHACKS

Application Portal


Email


Password

[Forgot Password?](#)

[Sign In](#)

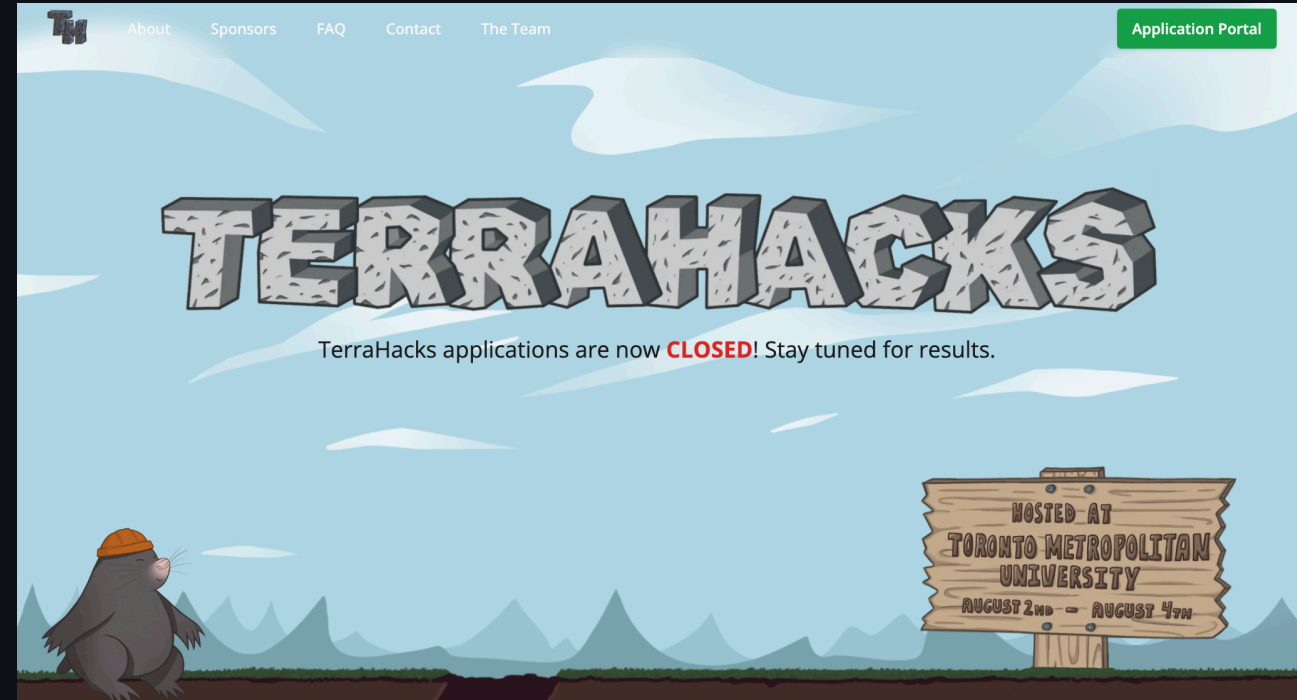
Don't have an account? [Sign up.](#)

 Continue with Google

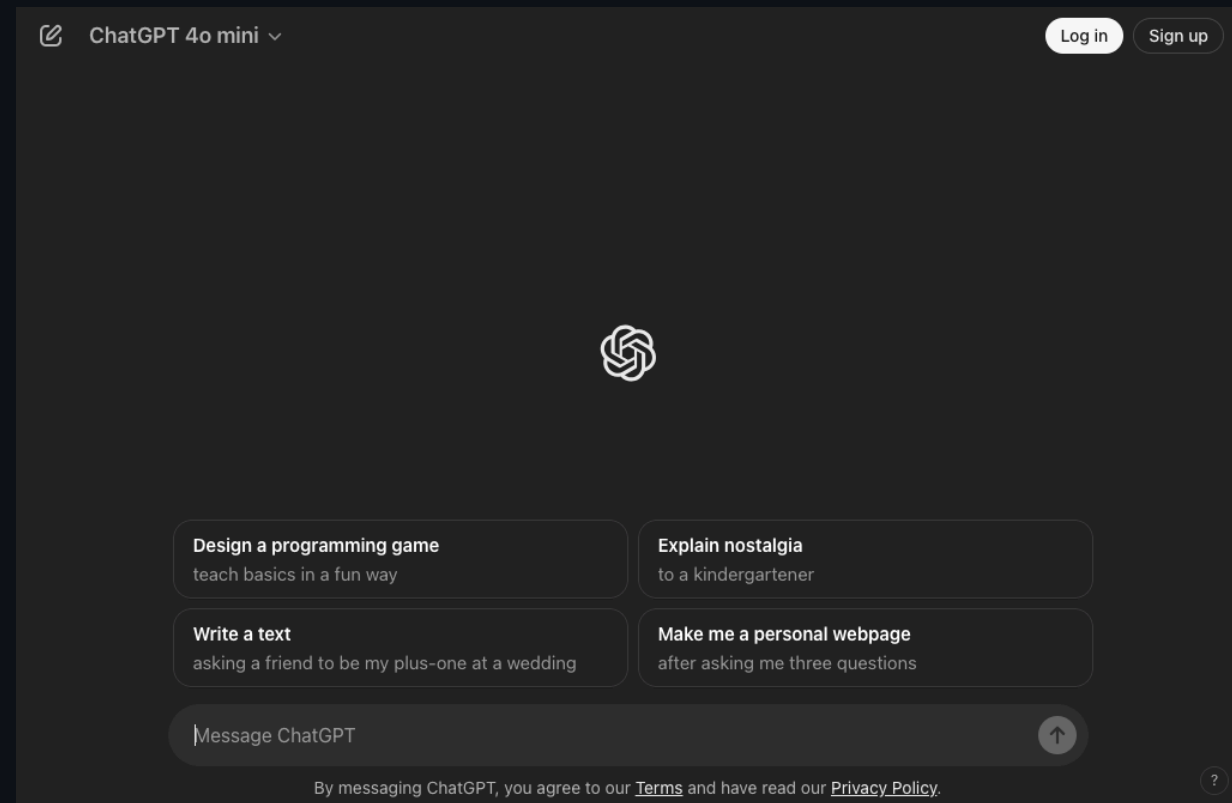
 Continue with GitHub

Having problems? Please contact us at contact@terrahacks.ca to get help.


TerraHacks Website



ChatGPT Website



Node.js Website

 [Learn](#) [About](#) [Download](#) [Blog](#) [Docs](#) [Certification](#) ↗

Start typing... ✖ ⌕ 🌙 🏠 🔍

Run JavaScript Everywhere

Node.js® is a free, open-source, cross-platform JavaScript runtime environment that lets developers create servers, web apps, command line tools and scripts.

Download Node.js (LTS) 📄

Downloads Node.js v20.16.0¹ with long-term support.
Node.js can also be installed via [package managers](#).

Want new features sooner? Get **Node.js v22.5.1¹** instead.

Create an HTTP Server Write Tests Read and Hash a File Streams Pipeline Wor

```
1 // server.mjs
2 import { createServer } from 'node:http';
3
4 const server = createServer((req, res) => {
5   res.writeHead(200, { 'Content-Type': 'text/plain' });
6   res.end('Hello World!\n');
7 });
8
9 // starts a simple http server locally on port 3000
10 server.listen(3000, '127.0.0.1', () => {
11   console.log('Listening on 127.0.0.1:3000');
12 });
13
14 // run with 'node server.mjs'
```

JavaScript


Copy to clipboard


Learn more what Node.js is able to offer with our [Learning materials](#).


Trademark Policy Privacy Policy Code of Conduct Security Policy



© OpenJS Foundation 🗣️ 📧 ✂️ 🌐 📄

Spotify Web Player

 Spotify

 Home

 Search

 Your Library 

Create your first playlist

It's easy, we'll help you

Create playlist

Let's find some podcasts to follow

We'll keep you updated on new episodes

Browse podcasts

Legal

Safety & Privacy Center


Privacy Policy



Cookies

About Ads

Accessibility

Cookies

 English


 

Sign up


Log in

Popular artists


Show all




Post Malone
Artist




Eminem
Artist




Taylor Swift
Artist



Billie Eilish
Artist




Drake
Artist




The Weeknd
Artist

Popular albums


Show all




The Death of Slim Shady (Coup De...
Eminem




HIT ME HARD AND SOFT
Billie Eilish




One Thing At A Time
Morgan Wallen



THE TORTURED POETS...
Taylor Swift




The Rise and Fall of a Midwest Princess
Chappell Roan





The Great American Bar Scene
Zach Bryan


Popular radio


Show all


 RADIO

 RADIO

 RADIO

 RADIO

 RADIO

 RADIO

Getting Started

What you will need

- Node.js (<https://nodejs.org>)
- npm (<https://npmjs.com>)

Check if Node.js is installed

```
node --version
```

Check if npm is installed

```
npm --version
```

Create a new Next.js project

```
npx create-next-app next-app
```

Note: npx comes with an npm installation

Select the following choices

- ✓ Would you like to use TypeScript? ... No
- ✓ Would you like to use ESLint? ... Yes
- ✓ Would you like to use Tailwind CSS? ... Yes
- ✓ Would you like to use `src/` directory? ... No
- ✓ Would you like to use App Router? (recommended) ... Yes
- ✓ Would you like to customize the default import alias (@/*)? ... No

Starting the development server

```
cd next-app
```

```
npm run dev
```

Project Structure

```
▼ NEXT-APP
  ▼ app
    ★ favicon.ico
    # globals.css
    JS layout.js
    JS page.js
    > node_modules
  ▼ public
    🖼️ next.svg
    🖼️ vercel.svg
    🌀 .eslintrc.json
    💎 .gitignore
    {} jsconfig.json
    JS next.config.mjs
    {} package-lock.json
    {} package.json
    JS postcss.config.mjs
    ⓘ README.md
    JS tailwind.config.js
```

Routing in Next.js

- File based routing

Next.js built-in components

<Link>

```
import Link from 'next/link'
```

```
<Link href="https://google.com">Google</Link>
```

```
<Link href="/dashboard">Dashboard</Link>
```


<Image>

```
import Image from 'next/image'
```

```
<Image src="/logo.png" alt="Logo" width={200} height={200} />
```

Client Components

```
'use client';

export default function ClientComponent() {
  return (
    <button onClick={console.log("Clicked")}>Click</button>
  )
}
```

useState Hook

```
'use client';

import { useState } from 'react'

export default function Counter() {
  const [count, setCount] = useState(0)

  return (
    <div>
      <p>{count}</p>
      <button onClick={() => setCount(count + 1)}>Increment</button>
    </div>
  )
}
```

useEffect Hook

```
'use client';

import { useEffect } from 'react'

export default function Counter() {
  useEffect(() => {
    console.log("Component mounted") // Log will be shown in the browser console
  }, [])

  return (
    <div>
      <p>Counter</p>
    </div>
  )
}
```

Server Components

Every component in Next.js is a server component by default

```
export default function ServerComponent() {  
  console.log("Server Component") // Log will be shown in the server console  
  return (  
    <div>  
      <p>Server Component</p>  
    </div>  
  )  
}
```

Props

Values passed from parent to child components


```
// Parent Component
import ChildComponent from './ChildComponent'

export default function ParentComponent() {
  return (
    <ChildComponent name="Hello, World!" />
  )
}
```

```
// Child Component
"use client";
export default function ChildComponent({ name }) {
  return (
    <button onClick={() => console.log(`${name} button clicked`)}>{name}</button>
  )
}
```

Dynamic Routes

app/profile/[name]/page.js

```
export default function Profile({ params }) {  
  return (  
    <div>  
      <p>Name: {params.name}</p>  
    </div>  
  )  
}
```

Data Fetching

client side data fetching

server side data fetching

Server Actions

Thank you!

Questions?

Resources

- [Next.js Documentation](#)