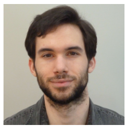
JUNE 21, 2021 / #REACT

How to Build a Portfolio Website with React



Reed Barger

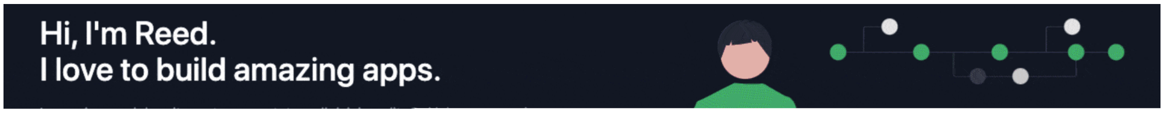
Today you're going to create one of the most important apps you can build for yourself: your developer portfolio.

Every React developer or web developer in general needs to be able to show off what they can do to any potential client or employer.

That's exactly what we're going to be building right now, with the help of a number of industry standard tools, including React, Tailwind CSS, and Netlify.

Let's get started!

What Will the Portfolio Look Like?



Hi, I'm Reed.
I love to build amazing apps.

This is the final version of the portfolio you will be building.

It will feature information about yourself, what projects you have made, what skills you've used in making those projects, and will include a contact form for clients or employers to reach out to you.

What Tools Will We Be Using?

- We will use React to create the app's user interface. It will allow us to compose each part of our landing page through reusable components and extend our app if we want to add additional features, such as a blog.
- To style our application, we will use Tailwind CSS. To give our app a professional appearance, Tailwind will allow us to easily apply multiple styles through combining classnames on our React elements.
- For pushing our app to the web, we will use the free service Netlify. It will serve our project on a custom domain to users very quickly with the help of a CDN (content delivery network).

How to Get Started

[You can download the starting files for our project here.]

When you grab the code, all you will have to do is drag your (unzipped) project folder into your code editor and run the command:

```
npm install
```

And you're good to go!

What Tools Do I Need to Build my Portfolio?

To go through the entire process of creating our app from start to deployment, you will need to have the following:

1. Node.js installed on your computer. You can download it at nodejs.org.
2. Git installed on your computer. You can download it at git-scm.com.
3. I would recommend using VS Code as your code editor. You can download it at code.visualstudio.com.
4. A free Netlify account at netlify.com.
5. A free GitHub account at github.com.

The benefit of using React is that we could expand our app to as many pages as we like, very simply, and add a lot of additional content.

However, since we're just working with one page, we can within our app component figure out the different components that we need very quickly. We will have a Navbar on top with all of the links to jump to different sections of our portfolio.

After that, we will include an about section, a section for our projects, testimonials, and finally our contact form.

This quick planning allows us to figure out what our components should be named and in what order. We can go ahead and add all of them to our App.js file (in src):

```
// src/App.js

import React from "react";

export default function App() {
  return (
    <main>
      <Navbar />
      <About />
      <Projects />
      <Skills />
      <Testimonials />
      <Contact />
    </main>
  );
}
```

ahead and create them.

Within our source (src) folder, we're going to create a folder called components with all of the files that we need:

```
my-portfolio
├── README.md
├── node_modules
├── package.json
├── .gitignore
├── public
│   ├── favicon.ico
│   ├── index.html
│   └── manifest.json
└── src
    ├── App.js
    ├── data.js
    ├── index.css
    ├── index.js
    └── components
        ├── About.js
        ├── Contact.js
        ├── Navbar.js
        ├── Projects.js
        ├── Skills.js
        └── Testimonials.js
```

Then we will create the basic structure of each React component and export it from that file with `export default` :

```
// src/components/About.js

export default function About() {}
```

And finally make sure to import it back in App.js:

```
// src/App.js

import React from "react";
import About from "./components/About";
import Contact from "./components/Contact";
import Navbar from "./components/Navbar";
import Projects from "./components/Projects";
import Skills from "./components/Skills";
import Testimonials from "./components/Testimonials";

export default function App() {
  return (
    <main>
      <Navbar />
      <About />
      <Projects />
      <Skills />
      <Testimonials />
      <Contact />
    </main>
  );
}
```

Note that there should be six components in total.

Intro to Tailwind CSS

Once that's done, we can start working with Tailwind CSS, in order to start to give our app a basic appearance.

The benefit of using Tailwind CSS is that we don't have to write any

For example, to give our portfolio a dark background with gray text applied to all of our child components, you can add the following classes to our `main` element:

```
// src/App.js

import React from "react";
import About from "./components/About";
import Contact from "./components/Contact";
import Navbar from "./components/Navbar";
import Projects from "./components/Projects";
import Skills from "./components/Skills";
import Testimonials from "./components/Testimonials";

export default function App() {
  return (
    <main className="text-gray-400 bg-gray-900 body-font">
      <Navbar />
      <About />
      <Projects />
      <Skills />
      <Testimonials />
      <Contact />
    </main>
  );
}
```

How to Build the About Component

Let's start on our first section, the about section. This will consist of a basic introduction to ourselves and what skills we specialize in.

It's also going to include some links to the contact form as well as our

To make these links work and to be able to jump to each section, we will set the `id` attribute of the projects section to "projects" and those of the contact section to "contact".

```
// src/components/About.js

import React from "react";

export default function About() {
  return (
    <section id="about">
      <div className="container mx-auto flex px-10 py-20 md:flex-row fle
        <div className="lg:flex-grow md:w-1/2 lg:pr-24 md:pr-16 flex fle
          <h1 className="title-font sm:text-4xl text-3xl mb-4 font-mediu
            Hi, I'm Reed.
            <br className="hidden lg:inline-block" />I love to build amaz
            apps.
          </h1>
          <p className="mb-8 leading-relaxed">
            Lorem ipsum dolor sit amet, consectetur adipisicing elit. Qui
            laborum quasi, incidunt dolore iste nostrum cupiditate volupt
            Laborum, voluptas natus?
          </p>
          <div className="flex justify-center">
            <a
              href="#contact"
              className="inline-flex text-white bg-green-500 border-0 py-
              Work With Me
            </a>
            <a
              href="#projects"
              className="ml-4 inline-flex text-gray-400 bg-gray-800 borde
              See My Past Work
            </a>
          </div>
        </div>
        <div className="lg:max-w-lg lg:w-full md:w-1/2 w-5/6">
          <img
        </div>
      </div>
    </section>
  );
}
```

For the image on the righthand side of the section, I am using an svg file from the `public` folder, coding.svg.

This image serves merely as a temporary placeholder. I would highly recommend using an actual image of yourself.

How to Build the Projects Component

Our projects section will consist of a `section` element with an `id` of "projects". This will feature a gallery of all the projects that we've built, which will include images.

It'll have the title of the project, along with the technologies we use to make it, and a link to it (if it is deployed).

```
// src/components/Projects.js

import { CodeIcon } from "@heroicons/react/solid";
import React from "react";
import { projects } from "../data";

export default function Projects() {
  return (
          <h1 className="sm:text-4xl text-3xl font-medium title-font mb-4
            Apps I've Built
          </h1>
          <p className="lg:w-2/3 mx-auto leading-relaxed text-base">
            Lorem ipsum, dolor sit amet consectetur adipisicing elit. Exp
            facilis repellat ab cupiditate alias vero aliquid obcaecati c
            fuga dolore.
          </p>
        </div>
        <div className="flex flex-wrap -m-4">
          {projects.map((project) => (
            <a
              href={project.link}
              key={project.image}
              className="sm:w-1/2 w-100 p-4">
              <div className="flex relative">
                <img
                  alt="gallery"
                  className="absolute inset-0 w-full h-full object-cover
                  src={project.image}
                />
                <div className="px-8 py-10 relative z-10 w-full border-4
                  <h2 className="tracking-widest text-sm title-font font-
                    {project.subtitle}
                  </h2>
                  <h1 className="title-font text-lg font-medium text-whit
                    {project.title}
                  </h1>
                  <p className="leading-relaxed">{project.description}</p
                </div>
              </div>
            </a>
          ))}
        </div>
      </div>
    </section>
  );
}
```

We are importing an array of projects from a data.js file in the same folder. There we are exporting an array of objects which each include an individual project's data:

```
// src/data.js

export const projects = [
  {
    title: "React Reserve",
    subtitle: "MERN Stack",
    description:
      "Lorem ipsum dolor sit amet consectetur adipisicing elit. Praesenti
    image: "./project-1.gif",
    link: "https://reactbootcamp.com",
  },
  {
    title: "React Tracks",
    subtitle: "React and Python",
    description:
      "Lorem ipsum dolor sit amet consectetur adipisicing elit. Praesenti
    image: "./project-2.gif",
    link: "https://reedbarger.com",
  },
  {
    title: "DevChat",
    subtitle: "React and Firebase",
    description:
      "Lorem ipsum dolor sit amet consectetur adipisicing elit. Praesenti
    image: "./project-3.gif",
    link: "https://jsbootcamp.com",
  },
  {
    title: "Epic Todo App",
    subtitle: "React Hooks",
    description:
      "Lorem ipsum dolor sit amet consectetur adipisicing elit. Praesenti
    image: "./project-4.gif",
    link: "https://pythonbootcamp.com",
```

How to Build the Skills Component

Let's fill out the section for all the skills and technologies that we know.

This will consist of a simple list of all of the major tools that we're familiar with and can use in our employers or clients projects.

Once again, we are going to import an array from our data folder. But this array consists of number of strings which represent each of the skills that we know such as JavaScript, React, and Node:

```
// src/components/Skills.js

import { BadgeCheckIcon, ChipIcon } from "@heroicons/react/solid";
import React from "react";
import { skills } from "../data";

export default function Skills() {
  return (
    <section id="skills">
      <div className="container px-5 py-10 mx-auto">
        <div className="text-center mb-20">
          <ChipIcon className="w-10 inline-block mb-4" />
          <h1 className="sm:text-4xl text-3xl font-medium title-font text
            Skills &amp; Technologies
          </h1>
          <p className="text-base leading-relaxed xl:w-2/4 lg:w-3/4 mx-au
            Lorem ipsum dolor sit amet consectetur, adipisicing elit. Nis
            ipsa delectus eum quo voluptas aspernatur accusantium distinc
            possimus est.
          </p>
```

```
              <div className="bg-gray-800 rounded flex p-4 h-full items-c
                <BadgeCheckIcon className="text-green-400 w-6 h-6 flex-sh
                <span className="title-font font-medium text-white">
                  {skill}
                </span>
              </div>
            </div>
          ))}
        </div>
      </div>
    </section>
  );
}
```

How to Build the Testimonials Component

In the Testimonials component, we are going to list a couple of testimonials maybe from past clients or people who are familiar with our work.

These are going to consist of a couple of cards that feature the testimonial itself as well as who it's from and the company that this person is from.

We are also importing a testimonials array with a number of objects that feature the quote, image, name, and company.

```
// src/components/Testimonials

import React from "react";
```

```
  return (
    <section id="testimonials">
      <div className="container px-5 py-10 mx-auto text-center">
        <UsersIcon className="w-10 inline-block mb-4" />
        <h1 className="sm:text-4xl text-3xl font-medium title-font text-w
          Client Testimonials
        </h1>
        <div className="flex flex-wrap m-4">
          {testimonials.map((testimonial) => (
            <div className="p-4 md:w-1/2 w-full">
              <div className="h-full bg-gray-800 bg-opacity-40 p-8 rounde
                <TerminalIcon className="block w-8 text-gray-500 mb-4" /
                <p className="leading-relaxed mb-6">{testimonial.quote}</
                <div className="inline-flex items-center">
                  <img
                    alt="testimonial"
                    src={testimonial.image}
                    className="w-12 rounded-full flex-shrink-0 object-co
                  />
                  <span className="flex-grow flex flex-col pl-4">
                    <span className="title-font font-medium text-white">
                      {testimonial.name}
                    </span>
                    <span className="text-gray-500 text-sm uppercase">
                      {testimonial.company}
                    </span>
                  </span>
                </div>
              </div>
            </div>
          ))}
        </div>
      </div>
    </section>
  );
}
```

form to allow potential employers to reach out to us.

This form will have 3 inputs: a name, email, and message input.

To receive these form submissions, we will use the tool Netlify Forms to very easily take care of saving those messages.

```
// src/components/Contact.js

import React from "react";

export default function Contact() {
  return (
    <section id="contact" className="relative">
      <div className="container px-5 py-10 mx-auto flex sm:flex-nowrap fl
        <div className="lg:w-2/3 md:w-1/2 bg-gray-900 rounded-lg overflow
          <iframe
            width="100%"
            height="100%"
            title="map"
            className="absolute inset-0"
            frameBorder={0}
            marginHeight={0}
            marginWidth={0}
            style={{ filter: "opacity(0.7)" }}
            src="https://www.google.com/maps/embed/v1/place?q=97+warren+s
          />
          <div className="bg-gray-900 relative flex flex-wrap py-6 rounde
            <div className="lg:w-1/2 px-6">
              <h2 className="title-font font-semibold text-white tracking
                ADDRESS
              </h2>
              <p className="mt-1">
                97 Warren St. <br />
                New York, NY 10007
              </p>
            </div>
              <a className="text-indigo-400 leading-relaxed">
                reedbarger@email.com
              </a>
              <h2 className="title-font font-semibold text-white tracking
                PHONE
              </h2>
              <p className="leading-relaxed">123-456-7890</p>
            </div>
          </div>
        </div>
        <form
          netlify
          name="contact"
          className="lg:w-1/3 md:w-1/2 flex flex-col md:ml-auto w-full md
          <h2 className="text-white sm:text-4xl text-3xl mb-1 font-medium
            Hire Me
          </h2>
          <p className="leading-relaxed mb-5">
            Lorem ipsum dolor sit amet consectetur, adipisicing elit. Ill
            suscipit officia aspernatur veritatis. Asperiores, aliquid?
          </p>
          <div className="relative mb-4">
            <label htmlFor="name" className="leading-7 text-sm text-gray-
              Name
            </label>
            <input
              type="text"
              id="name"
              name="name"
              className="w-full bg-gray-800 rounded border border-gray-7
            />
          </div>
          <div className="relative mb-4">
            <label htmlFor="email" className="leading-7 text-sm text-gray
              Email
            </label>
            <input
              type="email"
              id="email"
              name="email"
              className="w-full bg-gray-800 rounded border border-gray-7
            />
```

```
              className="leading-7 text-sm text-gray-400">
              Message
            </label>
            <textarea
              id="message"
              name="message"
              className="w-full bg-gray-800 rounded border border-gray-70
            />
          </div>
          <button
            type="submit"
            className="text-white bg-indigo-500 border-0 py-2 px-6 focus:
            Submit
          </button>
        </form>
      </div>
    </section>
  );
}
```

How to Embed a Google Maps Location

To the left of the form we will include a Google Maps embedded Google map of where we are located.

We can do so with the help of an online tool: embed-map.com. All you have to do is just enter your location and hit "Generate HTML code".

In the code we are given, don't copy all of the code, just the `src` attribute from the iframe element. We will replace that value with the default `src` value we have for our iframe.

to recognize a form as static HTML. Because our React app is controlled by JavaScript and doesn't consist of plain HTML, we need to add a hidden form to our index.html file in the public folder.

```html
<!-- public/index.html -->

<!DOCTYPE html>
<html lang="en">
  <head>
    <!-- head content skipped -->
  </head>
  <body>

  <form name="contact" netlify netlify-honeypot="bot-field" hidden>
    <input type="text" name="name" />
    <input type="email" name="email" />
    <textarea name="message"></textarea>
  </form>

    <noscript>You need to enable JavaScript to run this app.</noscript>
    <div id="root"></div>
  </body>
</html>
```

We need to hide this form, because it doesn't need to be seen by the user, just Netlify.

We'll give it the attribute of `hidden` as well as a `name` attribute that matches the JSX form in Contact.js. We also need to give it the `netlify` attribute so that Netlify Forms recognizes it. Finally, we need to include all of the same inputs as our JSX form: for name, email, message.

Once that's done, we'll head back to Contact.js. We're going to use JavaScript in order to submit this form.

First of all, we're going to create some dedicated state for each of the values that are typed in the form for name, email, and message:

```
const [name, setName] = React.useState("");
const [email, setEmail] = React.useState("");
const [message, setMessage] = React.useState("");
```

We will store what the user types in to each of the inputs in state with the help of the `onChange` handler.

To handle submission of the form, we will add the `onSubmit` prop to it. The function that will be called, `handleSubmit`, will make a post request to the endpoint "/" with all of our form data.

We will set the headers of the request to indicate that we are sending over form data. For the request body, we will include the form name as well as all of the form data from the `name`, `email`, and `message` state variables.

```
// src/components/Contact.js

import React from "react";

export default function Contact() {
  const [name, setName] = React.useState("");
  const [email, setEmail] = React.useState("");
```

```
      .map(
        (key) => encodeURIComponent(key) + "=" + encodeURIComponent(data
      )
      .join("&");
  }

  function handleSubmit(e) {
    e.preventDefault();
    fetch("/", {
      method: "POST",
      headers: { "Content-Type": "application/x-www-form-urlencoded" },
      body: encode({ "form-name": "contact", name, email, message }),
    })
      .then(() => alert("Message sent!"))
      .catch((error) => alert(error));
  }

  return (
    <section id="contact" className="relative">
      <div className="container px-5 py-10 mx-auto flex sm:flex-nowrap fl
        <div className="lg:w-2/3 md:w-1/2 bg-gray-900 rounded-lg overflow
          <iframe
            width="100%"
            height="100%"
            title="map"
            className="absolute inset-0"
            frameBorder={0}
            marginHeight={0}
            marginWidth={0}
            style={{ filter: "opacity(0.7)" }}
            src="https://www.google.com/maps/embed/v1/place?q=97+warren+s
          />
          <div className="bg-gray-900 relative flex flex-wrap py-6 rounde
            <div className="lg:w-1/2 px-6">
              <h2 className="title-font font-semibold text-white tracking
                ADDRESS
              </h2>
              <p className="mt-1">
                97 Warren St. <br />
                New York, NY 10007
              </p>
            </div>
              <a className="text-indigo-400 leading-relaxed">
                reedbarger@email.com
              </a>
              <h2 className="title-font font-semibold text-white tracking
                PHONE
              </h2>
              <p className="leading-relaxed">123-456-7890</p>
            </div>
          </div>
        </div>
        <form
          netlify
          name="contact"
          onSubmit={handleSubmit}
          className="lg:w-1/3 md:w-1/2 flex flex-col md:ml-auto w-full mt
          <h2 className="text-white sm:text-4xl text-3xl mb-1 font-medium
            Hire Me
          </h2>
          <p className="leading-relaxed mb-5">
            Lorem ipsum dolor sit amet consectetur, adipisicing elit. Il
            suscipit officia aspernatur veritatis. Asperiores, aliquid?
          </p>
          <div className="relative mb-4">
            <label htmlFor="name" className="leading-7 text-sm text-gray-
              Name
            </label>
            <input
              type="text"
              id="name"
              name="name"
              className="w-full bg-gray-800 rounded border border-gray-7
              onChange={(e) => setName(e.target.value)}
            />
          </div>
          <div className="relative mb-4">
            <label htmlFor="email" className="leading-7 text-sm text-gray
              Email
            </label>
            <input
              type="email"
              id="email"
              name="email"
```

```
          <div className="relative mb-4">
            <label
              htmlFor="message"
              className="leading-7 text-sm text-gray-400">
              Message
            </label>
            <textarea
              id="message"
              name="message"
              className="w-full bg-gray-800 rounded border border-gray-7
              onChange={(e) => setMessage(e.target.value)}
            />
          </div>
          <button
            type="submit"
            className="text-white bg-indigo-500 border-0 py-2 px-6 focus:
            Submit
          </button>
        </form>
      </div>
    </section>
  );
}
```

As you can see above, we are encoding the form data with a special `encode` function that you see here.

If the message is sent correctly, we will display an alert that says "Message sent". Otherwise if there is an error, we are going to alert the user of that error.

How to Build the Navbar Component

not be sticky on mobile devices.

Additionally, we want to include links to each of our relevant sections for our project skills testimonials and our contact form:

```
// src/components/Navbar.js

import { ArrowRightIcon } from "@heroicons/react/solid";
import React from "react";

export default function Navbar() {
  return (
    <header className="bg-gray-800 md:sticky top-0 z-10">
      <div className="container mx-auto flex flex-wrap p-5 flex-col md:fl
        <a className="title-font font-medium text-white mb-4 md:mb-0">
          <a href="#about" className="ml-3 text-xl">
            Reed Barger
          </a>
        </a>
        <nav className="md:mr-auto md:ml-4 md:py-1 md:pl-4 md:border-l mc
          <a href="#projects" className="mr-5 hover:text-white">
            Past Work
          </a>
          <a href="#skills" className="mr-5 hover:text-white">
            Skills
          </a>
          <a href="#testimonials" className="mr-5 hover:text-white">
            Testimonials
          </a>
        </nav>
        <a
          href="#contact"
          className="inline-flex items-center bg-gray-800 border-0 py-1 p
          Hire Me
          <ArrowRightIcon className="w-4 h-4 ml-1" />
        </a>
      </div>
    </header>
```

How does this stick to top of the page on a larger device? With the help of the class `md:sticky` on our `header` element.

This class means that it will have the style rule `position: sticky;` applied starting on a medium-sized breakpoint (768px).

How to Deploy Your Portfolio

Now to make our portfolio live, we need to push our application to GitHub.

If you're not familiar with Git and GitHub, I would take a little while just to learn how to push your code to your GitHub account for the first time. It's an essential skill for any developer to know.

Once you're familiar with this process, we can first create a new Github repository. After that, we will run `git add .`, `git commit -m "Deploy"`, create our git remote, and `git push -u origin master`.

Once our project is on GitHub, we can head over to Netlify and select the option "Choose Site from Git". Then we will choose GitHub for our continuous deployment, and pick the GitHub repository to which we just pushed our code.

After that, our project will be automatically deployed to the web!

shows off all of your projects and skills to potential employers.

The next step to take would be to set up a custom domain, preferably with your name (i.e. reedbarger.com). Since Netlify includes a DNS you can easily set up a custom domain with them.

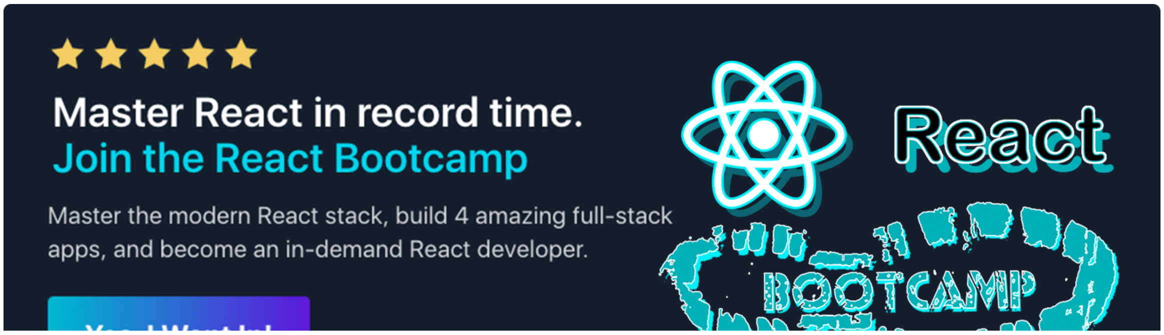
Look into maybe adding a blog to your React app to show off even more of your developer knowledge to potential employers.

Make your personal portfolio an expression of yourself and what you are passionate about as a developer and you'll have success!

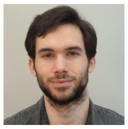
Want the #1 Way to Learn React?

The React Bootcamp takes everything you should know about learning React and bundles it into one comprehensive package, including videos, cheatsheets, plus special bonuses.

Gain the insider information **100s** of developers have already used to master React, find their dream jobs, and take control of their future:


★★★★★
Master React in record time.
Join the React Bootcamp
Master the modern React stack, build 4 amazing full-stack apps, and become an in-demand React developer.
React
BOOTCAMP

Forum Donate



Reed Barger

React developer who loves to make incredible apps. Showing you how at ReactBootcamp.com

If this article was helpful, tweet it.

Learn to code for free. freeCodeCamp's open source curriculum has helped more than 40,000 people get jobs as developers.

Get started

Trending Guides

Discord Won't Open

C vs C++

Forum Donate

File Explorer Error

Python Find in List

Functions in Python

Python Reverse List

Create a Table in SQL

List Index Out of Range

Python String.Replace()

How to Open Task Manager

Design Thinking Explained

Learn Programming Courses

Make a Transparent Taskbar

Install Ethernet Driver PC

What is Coding?

Text Box in HTML

Meta Tag in HTML

Append in Python

Python Not Equal

Linux awk Command

JS String Contains

Default Constructor in Java

Stuck Win 10 Hard Drive

Color Codes for Grey Palette

Binary Search Tree Traversal

RTC Connecting Discord Fix

Our Nonprofit

About Alumni Network Open Source Shop Support Sponsors Academic Honesty

Code of Conduct Privacy Policy Terms of Service Copyright Policy