Completing my Portfolio

The tech stack being used is as follows:

**🌟 Varun's Portfolio Tech Stack (2025)**

**🖼️ Frontend**

| **Tech** | **Purpose** |
| --- | --- |
| **React** | Core UI framework (component-based, powerful, flexible) |
| **Vite** | Super-fast build tool and dev server (replaces Create React App) |
| **Tailwind CSS** | Utility-first CSS framework (no need to write manual CSS) |
| **Framer Motion** | Animation library for smooth transitions, scroll/hover effects |
| **Spline** | Embeddable 3D objects and interactive visuals |
| **Heroicons / Lucide** | Modern, open-source icon sets for buttons and UI elements |
| **JSX + ES6** | The language you’ll write everything in (JavaScript + XML style markup) |

**🧠 Content Management**

| **Option** | **Purpose** |
| --- | --- |
| **Static JS/JSON files** | Project list, skills, experience (easy, local, fast) |
| *(Later)* **Sanity / Contentful** | Headless CMS to edit content without code |
| *(Optional)* **MDX** | If you want blog posts or long-form content with embedded components |

**🪄 Interactivity / Animations**

| **Tool** | **Purpose** |
| --- | --- |
| **Framer Motion** | Animate page transitions, cards, hover effects, reveals |
| **Spline** | Drag-n-drop 3D editor; export as React components or <iframe> |
| **React State / Props** | Control interactivity, animations, and UI states |

**🌐 Backend (Optional)**

If you later want form submissions, authentication, or dynamic content:

| **Tool** | **Purpose** |
| --- | --- |
| **Node.js + Express** | Backend server with API endpoints |
| **Supabase / Firebase** | Authentication + Realtime DB (no backend code needed) |
| **Formspree / EmailJS** | Serverless contact form handling |
| **Airtable / Notion API** | Low-code, spreadsheet-style backend for content or testimonials |

**🚀 Deployment**

| **Platform** | **Features** |
| --- | --- |
| **Vercel** | Seamless deployment from GitHub, optimized for React + Vite |
| *(Alt)* Netlify | Great for JAMstack, also works well with Tailwind |
| **Custom Domain** | Buy from Namecheap/Google Domains and link to Vercel in 2 clicks |

**🔧 Tooling & Dev Setup**

| **Tool** | **Role** |
| --- | --- |
| **Node.js** | Runtime that powers npm, dev server, and build tools |
| **npm** | Package manager to install Tailwind, React, Framer Motion, etc. |
| **Git + GitHub Desktop** | Version control and online repo hosting |
| **.gitignore** | Keeps build files and secrets out of GitHub |
| **VS Code** | Recommended editor (with Tailwind and React plugins) |

**📝 License**

| **Choice** | **Why** |
| --- | --- |
| **MIT License** | Allows others to view/learn from your code, with credit |
| *(Optional)* No License | If you want to keep it all-rights-reserved (not recommended) |

Resources:

* <https://web.dev/learn/css/animations>
* [React Demo 1](https://www.youtube.com/watch?v=ifOJ0R5UQOc&t=3390s&ab_channel=PedroTech)
* <https://cssreference.io/animations/>

**Steps:**

1. Create a local Git repository using GitHub Desktop, and add Node to the *.gitignore* file. In addition to this, I selected the MIT License to protect my work, while also allowing others to learn from it.
2. Initialized Vite and React with the following statement:
   1. npm create vite@latest . -- --template react
   2. npm install
   3. npm run dev (to run code locally)
3. Installing Tailwind CSS and additional packages:
   1. <https://tailwindcss.com/docs/installation/using-vite>
   2. Have your plugins like this in the vite.config.js:
   3. A screen shot of a computer code

      AI-generated content may be incorrect.
   4. Make sure it runs npm run dev with no issues
   5. Install framer motion with npm install framer-motion
   6. lucide-react for certain icons
   7. react-router-dom for routes to various pages, if making a multipaged site
   8. tailwind-merge to create a util function to combine class names
   9. @radix-ui/react-toast class-variance-authority clsx are packages used to allow for toaster messages (pop-up prompts)
4. Adding a not found page, and home page, import BrowserRouter, Route, and Routes, this will allow you to jump from page to page: [React Demo 1](https://www.youtube.com/watch?v=ifOJ0R5UQOc&t=3390s&ab_channel=PedroTech)
   1. To add the spline model, simply copy the embed public link from spline and add the iframe to the div, but add the class name className="w-screen h-screen overflow-hidden" to the div and className="w-full h-full" allowFullScreen settings to the iframe to ensure the full page is taken up.
5. Create a Resume page similar way of NotFound, and add the iframe container to handle the resume, with the resume stored in public, as that is where all static files should be
6. Delete references to the default App.css and delete all template code given by vite, from there begin to configure your styling and animations needed for the website in index.css, this is dependent on personal preferences and needs of the project.
7. In components create a .jsx file “Navbar” which will feature the navigation or jump bar for the portfolio website.