

Software Engineer Portfolio Website Documentation

Project Overview

A professional portfolio website built using **React** and **TypeScript**. Its purpose is to showcase your skills, highlight your best projects, provide insights into your experience, and offer contact options for recruiters and collaborators.

1. Goals

- Present yourself as a qualified software engineer.
 - Showcase your technical projects with detailed case studies.
 - Allow employers to view, download, or share your resume.
 - Provide a clean and professional UI with strong UX.
 - Ensure SEO and accessibility for maximum visibility.
-

2. Tech Stack

- **Frontend:** React + TypeScript
 - **Styling:** Tailwind CSS or CSS Modules
 - **Routing:** React Router or Next.js-like structure (if using React SPA)
 - **Build Tool:** Vite or Create React App
 - **Hosting:** Vercel, Netlify, or GitHub Pages
 - **Contact Handling:** FormSubmit, serverless function, or custom backend
-

3. Essential Pages & Routes

- **Home / Hero** – Introduction, tagline, highlight buttons.
 - **About** – Bio, skills, professional journey.
 - **Projects** – Project gallery.
 - **Project Details** – Case studies.
 - **Resume** – View & download PDF.
 - **Contact** – Contact form & social links.
 - **Blog (Optional)** – Technical articles.
-

4. Components Breakdown

UI Components

- Button
- Input
- Textarea
- Tag/Badge
- Card
- Modal
- Navbar
- Footer

Layout Components

- Container
- Section
- Grid
- PageHeader

Portfolio Components

- HeroSection
- AboutSection
- SkillList
- ProjectCard
- ProjectGallery
- ProjectDetail
- ResumeViewer
- ContactForm

5. Project Data Structure

```
export interface Project {  
  id: string;  
  title: string;  
  slug: string;  
  shortDescription: string;  
  longDescription: string;  
  thumbnail: string;  
  images?: string[];  
  tech: string[];  
  repo?: string;  
  demo?: string;  
  role?: string;
```

```
    highlights?: string[];  
}
```

6. Case Study Template

Project Title

One-line Summary

Context & Role

Your role, responsibilities, and purpose of the project.

Problem Statement

What issue the project solves.

Goals

Measurable objectives.

Solution Overview

Architecture, UI, features, and technologies used.

Challenges & Decisions

Trade-offs, design decisions, optimization methods.

Results

Outcomes, performance improvements, impact.

Learnings

Technical and soft-skill takeaways.

Links

- GitHub Repo
 - Live Demo
-

7. Accessibility Checklist

- Semantic HTML
 - Alt text for images
 - Keyboard navigation
 - Proper color contrast
 - ARIA labels where necessary
-

8. SEO Checklist

- Unique title + meta description per page
 - Open Graph tags
 - Sitemap and robots.txt
 - Optimized images
 - Fast loading performance
-

9. Deployment Checklist

- Domain configured
 - SSL enabled
 - Contact form working
 - Resume downloadable
 - Lighthouse score above 90
 - Mobile responsiveness verified
-

10. Folder Structure Example (React + TS)

```
src/
  components/
  pages/
  layouts/
  data/
  hooks/
  styles/
  assets/
  types/
```

11. Deliverables

- Fully built portfolio website
 - Professional resume PDF
 - High-quality project screenshots
 - 4–8 full case studies
 - SEO-friendly metadata
-

This document outlines all required planning and structure for building your React + TypeScript portfolio website. You can now export and use it for development.