**Detailed Portfolio Development Plan with Git Commits**

**Phase 1: Project Setup & Foundation (Days 1-2)**

**Day 1: Initial Setup**

**1.1 Create Next.js Project**

bash

npx create-next-app@latest my-portfolio --no-typescript --tailwind --eslint --app --src-dir --import-alias "@/\*"

cd my-portfolio

**1.2 Install Dependencies**

bash

npm install framer-motion clsx tailwind-merge lucide-react

npm install @radix-ui/react-icons

**1.3 Initialize ShadCN UI**

bash

npx shadcn-ui@latest init

* Choose: yes for TypeScript? → no
* Style: new-york
* Base color: slate
* CSS variables: yes
* React server component? → yes
* Import alias: @/\*

**1.4 Add ShadCN Components**

bash

npx shadcn-ui@latest add button

npx shadcn-ui@latest add card

npx shadcn-ui@latest add dropdown-menu

npx shadcn-ui@latest add sheet

npx shadcn-ui@latest add avatar

npx shadcn-ui@latest add badge

npx shadcn-ui@latest add switch

npx shadcn-ui@latest add input

npx shadcn-ui@latest add textarea

**Commit 1: "feat: initial project setup with Next.js, Tailwind, and ShadCN"**

bash

git add .

git commit -m "feat: initial project setup with Next.js, Tailwind, and ShadCN

- Initialize Next.js 14 app with App Router

- Configure Tailwind CSS with dark mode

- Setup ShadCN UI components system

- Install Framer Motion for animations

- Add project structure foundation"

**Day 2: Theme System & Configuration**

**2.1 Update Tailwind Config** (tailwind.config.js)

javascript

*/\*\* @type {import('tailwindcss').Config} \*/*

module.exports = {

darkMode: ["class"],

content: [

'./src/\*\*/\*.{js,ts,jsx}',

],

theme: {

container: {

center: true,

padding: "2rem",

screens: {

"2xl": "1400px",

},

},

extend: {

colors: {

border: "hsl(var(--border))",

background: "hsl(var(--background))",

foreground: "hsl(var(--foreground))",

primary: {

DEFAULT: "hsl(var(--primary))",

foreground: "hsl(var(--primary-foreground))",

},

*// Add other color variables...*

},

animation: {

"fade-up": "fadeUp 0.5s ease-out",

"fade-in": "fadeIn 0.3s ease-in",

},

keyframes: {

fadeUp: {

"0%": { opacity: 0, transform: "translateY(20px)" },

"100%": { opacity: 1, transform: "translateY(0)" },

},

fadeIn: {

"0%": { opacity: 0 },

"100%": { opacity: 1 },

},

},

},

},

plugins: [],

}

**2.2 Create Theme Provider** (src/contexts/ThemeContext.jsx)

javascript

"use client"

import { createContext, useContext, useEffect, useState } from 'react'

const ThemeContext = createContext()

export function ThemeProvider({ children }) {

const [theme, setTheme] = useState('system')

useEffect(() => {

const savedTheme = localStorage.getItem('theme') || 'system'

setTheme(savedTheme)

}, [])

useEffect(() => {

const root = window.document.documentElement

root.classList.remove('light', 'dark')

if (theme === 'system') {

const systemTheme = window.matchMedia('(prefers-color-scheme: dark)').matches ? 'dark' : 'light'

root.classList.add(systemTheme)

} else {

root.classList.add(theme)

}

localStorage.setItem('theme', theme)

}, [theme])

return (

<ThemeContext.Provider value={{ theme, setTheme }}>

{children}

</ThemeContext.Provider>

)

}

export const useTheme = () => {

const context = useContext(ThemeContext)

if (!context) {

throw new Error('useTheme must be used within a ThemeProvider')

}

return context

}

**2.3 Create Utility Functions** (src/lib/utils.js)

javascript

import { clsx } from 'clsx'

import { twMerge } from 'tailwind-merge'

export function cn(...inputs) {

return twMerge(clsx(inputs))

}

export function formatDate(date) {

return new Date(date).toLocaleDateString('en-US', {

month: 'long',

year: 'numeric'

})

}

**2.4 Update Root Layout** (src/app/layout.jsx)

javascript

import { Inter } from 'next/font/google'

import './globals.css'

import { ThemeProvider } from '@/contexts/ThemeContext'

const inter = Inter({ subsets: ['latin'] })

export const metadata = {

title: 'Your Name - Portfolio',

description: 'Personal portfolio showcasing projects and experience',

}

export default function RootLayout({ children }) {

return (

<html lang="en" suppressHydrationWarning>

<body className={inter.className}>

<ThemeProvider>

{children}

</ThemeProvider>

</body>

</html>

)

}

**Commit 2: "feat: implement theme system with context provider"**

bash

git add .

git commit -m "feat: implement theme system with context provider

- Create ThemeContext with system/light/dark modes

- Add localStorage persistence for theme preference

- Extend Tailwind config with custom animations and CSS variables

- Create utility functions for className merging

- Update root layout with ThemeProvider"

**Phase 2: Core Components & Layout (Days 3-5)**

**Day 3: Layout Components**

**3.1 Create Navigation Component** (src/components/layout/Header.jsx)

javascript

"use client"

import { useState } from 'react'

import Link from 'next/link'

import { useTheme } from '@/contexts/ThemeContext'

import { Button } from '@/components/ui/button'

import { Sheet, SheetContent, SheetTrigger } from '@/components/ui/sheet'

import { Moon, Sun, Monitor, Menu } from 'lucide-react'

export default function Header() {

const [isOpen, setIsOpen] = useState(false)

const { theme, setTheme } = useTheme()

const navigation = [

{ name: 'Home', href: '#home' },

{ name: 'About', href: '#about' },

{ name: 'Projects', href: '#projects' },

{ name: 'Experience', href: '#experience' },

{ name: 'Contact', href: '#contact' },

]

return (

<header className="fixed top-0 z-50 w-full border-b bg-background/80 backdrop-blur-md">

<nav className="container flex h-16 items-center justify-between">

<div className="font-bold text-xl">YourName</div>

{*/\* Desktop Navigation \*/*}

<div className="hidden md:flex items-center gap-6">

{navigation.map((item) => (

<Link

key={item.name}

href={item.href}

className="text-sm font-medium transition-colors hover:text-primary"

>

{item.name}

</Link>

))}

{*/\* Theme Toggle \*/*}

<div className="flex items-center gap-1 border rounded-lg p-1">

<Button

variant={theme === 'light' ? 'default' : 'ghost'}

size="sm"

onClick={() => setTheme('light')}

>

<Sun className="h-4 w-4" />

</Button>

<Button

variant={theme === 'dark' ? 'default' : 'ghost'}

size="sm"

onClick={() => setTheme('dark')}

>

<Moon className="h-4 w-4" />

</Button>

<Button

variant={theme === 'system' ? 'default' : 'ghost'}

size="sm"

onClick={() => setTheme('system')}

>

<Monitor className="h-4 w-4" />

</Button>

</div>

</div>

{*/\* Mobile Navigation \*/*}

<Sheet open={isOpen} onOpenChange={setIsOpen}>

<SheetTrigger asChild>

<Button variant="ghost" size="sm" className="md:hidden">

<Menu className="h-5 w-5" />

</Button>

</SheetTrigger>

<SheetContent>

<div className="flex flex-col gap-4 mt-8">

{navigation.map((item) => (

<Link

key={item.name}

href={item.href}

className="text-lg font-medium py-2"

onClick={() => setIsOpen(false)}

>

{item.name}

</Link>

))}

</div>

</SheetContent>

</Sheet>

</nav>

</header>

)

}

**3.2 Create Footer Component** (src/components/layout/Footer.jsx)

javascript

import Link from 'next/link'

import { Github, Linkedin, Mail, Twitter } from 'lucide-react'

export default function Footer() {

const socialLinks = [

{ icon: Github, href: 'https://github.com', label: 'GitHub' },

{ icon: Linkedin, href: 'https://linkedin.com', label: 'LinkedIn' },

{ icon: Twitter, href: 'https://twitter.com', label: 'Twitter' },

{ icon: Mail, href: 'mailto:your@email.com', label: 'Email' },

]

return (

<footer className="border-t py-8">

<div className="container">

<div className="flex flex-col md:flex-row justify-between items-center gap-4">

<div className="text-sm text-muted-foreground">

© 2024 Your Name. All rights reserved.

</div>

<div className="flex gap-4">

{socialLinks.map((social) => (

<Link

key={social.label}

href={social.href}

className="text-muted-foreground hover:text-foreground transition-colors"

aria-label={social.label}

>

<social.icon className="h-5 w-5" />

</Link>

))}

</div>

</div>

</div>

</footer>

)

}

**Commit 3: "feat: create layout components with responsive navigation"**

bash

git add .

git commit -m "feat: create layout components with responsive navigation

- Implement fixed header with backdrop blur

- Add desktop and mobile navigation with sheet component

- Create theme toggle with light/dark/system options

- Build footer with social links

- Make layout fully responsive"

**Day 4: Animation Components**

**4.1 Create Reusable Animation Components** (src/components/animations/FadeUp.jsx)

javascript

"use client"

import { motion } from 'framer-motion'

import { useInView } from 'framer-motion'

import { useRef } from 'react'

export function FadeUp({ children, delay = 0, duration = 0.5, className }) {

const ref = useRef(null)

const isInView = useInView(ref, { once: true, margin: '-50px' })

return (

<motion.div

ref={ref}

initial={{ opacity: 0, y: 30 }}

animate={isInView ? { opacity: 1, y: 0 } : { opacity: 0, y: 30 }}

transition={{ duration, delay, ease: 'easeOut' }}

className={className}

>

{children}

</motion.div>

)

}

**4.2 Stagger Children Animation** (src/components/animations/StaggerChildren.jsx)

javascript

"use client"

import { motion } from 'framer-motion'

import { useInView } from 'framer-motion'

import { useRef } from 'react'

export function StaggerChildren({ children, className, delay = 0 }) {

const ref = useRef(null)

const isInView = useInView(ref, { once: true, margin: '-50px' })

const container = {

hidden: { opacity: 0 },

show: {

opacity: 1,

transition: {

staggerChildren: 0.2,

delayChildren: delay,

}

}

}

return (

<motion.div

ref={ref}

variants={container}

initial="hidden"

animate={isInView ? "show" : "hidden"}

className={className}

>

{children}

</motion.div>

)

}

**4.3 Typewriter Component** (src/components/animations/Typewriter.jsx)

javascript

"use client"

import { motion } from 'framer-motion'

export function Typewriter({ text, className, delay = 0 }) {

const sentence = {

hidden: { opacity: 1 },

visible: {

opacity: 1,

transition: {

delay,

staggerChildren: 0.05,

}

}

}

const letter = {

hidden: { opacity: 0, y: 50 },

visible: {

opacity: 1,

y: 0,

}

}

return (

<motion.h1

className={className}

variants={sentence}

initial="hidden"

animate="visible"

>

{text.split('').map((char, index) => (

<motion.span key={`${char}-${index}`} variants={letter}>

{char}

</motion.span>

))}

</motion.h1>

)

}

**Commit 4: "feat: implement reusable animation components"**

bash

git add .

git commit -m "feat: implement reusable animation components

- Create FadeUp component for scroll-triggered animations

- Build StaggerChildren for sequenced animations

- Develop Typewriter component for text effects

- Add useInView hooks for scroll-based triggers

- Optimize animations with performance in mind"

**Phase 3: Portfolio Sections Development (Days 6-10)**

**Day 5: Hero Section**

**5.1 Create Hero Section** (src/components/sections/Hero.jsx)

javascript

"use client"

import { Button } from '@/components/ui/button'

import { FadeUp } from '@/components/animations/FadeUp'

import { Typewriter } from '@/components/animations/Typewriter'

import { Download, Github, Linkedin } from 'lucide-react'

export default function Hero() {

return (

<section id="home" className="min-h-screen flex items-center pt-16">

<div className="container">

<div className="grid md:grid-cols-2 gap-12 items-center">

{*/\* Text Content \*/*}

<div className="space-y-6">

<FadeUp>

<p className="text-lg text-primary font-medium">Hello, I'm</p>

</FadeUp>

<Typewriter

text="Your Name"

className="text-4xl md:text-6xl font-bold tracking-tight"

/>

<FadeUp delay={0.2}>

<h2 className="text-2xl md:text-4xl text-muted-foreground font-semibold">

Full Stack Developer

</h2>

</FadeUp>

<FadeUp delay={0.4}>

<p className="text-lg text-muted-foreground max-w-md">

I create amazing web experiences using modern technologies.

Passionate about clean code, user experience, and bringing ideas to life.

</p>

</FadeUp>

<FadeUp delay={0.6}>

<div className="flex gap-4 flex-wrap">

<Button size="lg" className="gap-2">

<Download className="h-4 w-4" />

Download CV

</Button>

<Button variant="outline" size="lg" className="gap-2">

<Github className="h-4 w-4" />

GitHub

</Button>

<Button variant="outline" size="lg" className="gap-2">

<Linkedin className="h-4 w-4" />

LinkedIn

</Button>

</div>

</FadeUp>

</div>

{*/\* Image/Graphic \*/*}

<FadeUp delay={0.8}>

<div className="relative">

<div className="w-80 h-80 mx-auto bg-gradient-to-br from-primary to-primary/50 rounded-full flex items-center justify-center">

<div className="w-72 h-72 bg-background rounded-full flex items-center justify-center">

<span className="text-4xl">👨‍💻</span>

</div>

</div>

{*/\* Floating elements \*/*}

<div className="absolute -top-4 -right-4 w-8 h-8 bg-primary rounded-full animate-bounce" />

<div className="absolute -bottom-4 -left-4 w-6 h-6 bg-primary rounded-full animate-pulse" />

</div>

</FadeUp>

</div>

</div>

</section>

)

}

**Commit 5: "feat: create animated hero section with typewriter effect"**

bash

git add .

git commit -m "feat: create animated hero section with typewriter effect

- Implement hero section with grid layout

- Add typewriter animation for name

- Include fade-up animations for all elements

- Create call-to-action buttons with icons

- Add decorative floating elements"

**Day 6: About Section**

**6.1 Create About Section** (src/components/sections/About.jsx)

javascript

"use client"

import { FadeUp } from '@/components/animations/FadeUp'

import { StaggerChildren } from '@/components/animations/StaggerChildren'

import { Card, CardContent } from '@/components/ui/card'

import { Code2, Palette, Database, Smartphone } from 'lucide-react'

export default function About() {

const skills = [

{

icon: Code2,

title: 'Frontend',

description: 'React, Next.js, TypeScript',

items: ['React', 'Next.js', 'JavaScript', 'TypeScript', 'HTML/CSS']

},

{

icon: Database,

title: 'Backend',

description: 'Node.js, Python, Databases',

items: ['Node.js', 'Python', 'MongoDB', 'PostgreSQL', 'REST APIs']

},

{

icon: Palette,

title: 'Design',

description: 'UI/UX, Responsive Design',

items: ['Figma', 'Tailwind CSS', 'Framer Motion', 'ShadCN UI']

},

{

icon: Smartphone,

title: 'Mobile',

description: 'Cross-platform Development',

items: ['React Native', 'Flutter', 'Expo']

}

]

return (

<section id="about" className="py-20">

<div className="container">

<FadeUp>

<h2 className="text-3xl md:text-4xl font-bold text-center mb-4">About Me</h2>

</FadeUp>

<FadeUp delay={0.2}>

<p className="text-xl text-muted-foreground text-center max-w-2xl mx-auto mb-12">

Passionate developer with X years of experience creating digital solutions

that make a difference.

</p>

</FadeUp>

<div className="grid lg:grid-cols-2 gap-12 items-start">

{*/\* Personal Story \*/*}

<div className="space-y-6">

<FadeUp>

<h3 className="text-2xl font-semibold">My Journey</h3>

</FadeUp>

<FadeUp delay={0.1}>

<p className="text-muted-foreground">

I started my programming journey in 20XX and have been passionate about

creating amazing web experiences ever since. I love turning complex

problems into simple, beautiful designs.

</p>

</FadeUp>

<FadeUp delay={0.2}>

<p className="text-muted-foreground">

When I'm not coding, you can find me exploring new technologies,

contributing to open-source projects, or enjoying the great outdoors.

</p>

</FadeUp>

</div>

{*/\* Skills Grid \*/*}

<StaggerChildren className="grid sm:grid-cols-2 gap-6">

{skills.map((skill, index) => (

<motion.div

key={skill.title}

variants={{

hidden: { opacity: 0, y: 20 },

show: { opacity: 1, y: 0 }

}}

>

<Card className="h-full hover:shadow-lg transition-shadow">

<CardContent className="p-6">

<div className="flex items-center gap-3 mb-3">

<div className="p-2 bg-primary/10 rounded-lg">

<skill.icon className="h-5 w-5 text-primary" />

</div>

<div>

<h4 className="font-semibold">{skill.title}</h4>

<p className="text-sm text-muted-foreground">{skill.description}</p>

</div>

</div>

<div className="flex flex-wrap gap-2">

{skill.items.map((item) => (

<span

key={item}

className="px-2 py-1 bg-secondary text-secondary-foreground rounded-md text-xs"

>

{item}

</span>

))}

</div>

</CardContent>

</Card>

</motion.div>

))}

</StaggerChildren>

</div>

</div>

</section>

)

}

**Commit 6: "feat: build about section with skills grid and personal story"**

bash

git add .

git commit -m "feat: build about section with skills grid and personal story

- Create about section with two-column layout

- Implement skills grid with animated cards

- Add personal journey description

- Include skill badges with hover effects

- Use stagger animations for skills grid"

**Day 7: Projects Section**

**7.1 Create Projects Data** (src/constants/data.js)

javascript

export const projects = [

{

id: 1,

title: "E-Commerce Platform",

description: "A full-stack e-commerce solution with modern features",

image: "/images/project-1.jpg",

technologies: ["Next.js", "TypeScript", "Stripe", "MongoDB"],

githubUrl: "https://github.com",

liveUrl: "https://example.com",

featured: true

},

{

id: 2,

title: "Task Management App",

description: "Collaborative task management with real-time updates",

technologies: ["React", "Node.js", "Socket.io", "PostgreSQL"],

githubUrl: "https://github.com",

liveUrl: "https://example.com",

featured: true

},

*// Add more projects...*

]

export const experiences = [

{

id: 1,

company: "Tech Company",

position: "Senior Developer",

period: "2022 - Present",

description: "Lead development of web applications...",

technologies: ["React", "Node.js", "AWS"]

},

*// Add more experiences...*

]

**7.2 Create Projects Section** (src/components/sections/Projects.jsx)

javascript

"use client"

import { useState } from 'react'

import { FadeUp } from '@/components/animations/FadeUp'

import { StaggerChildren } from '@/components/animations/StaggerChildren'

import { Card, CardContent } from '@/components/ui/card'

import { Button } from '@/components/ui/button'

import { Badge } from '@/components/ui/badge'

import { projects } from '@/constants/data'

import { ExternalLink, Github, ArrowRight } from 'lucide-react'

export default function Projects() {

const [filter, setFilter] = useState('all')

const filteredProjects = filter === 'all'

? projects

: projects.filter(project => project.technologies.includes(filter))

const technologies = [...new Set(projects.flatMap(p => p.technologies))]

return (

<section id="projects" className="py-20 bg-muted/50">

<div className="container">

<FadeUp>

<h2 className="text-3xl md:text-4xl font-bold text-center mb-4">Featured Projects</h2>

</FadeUp>

<FadeUp delay={0.2}>

<p className="text-xl text-muted-foreground text-center max-w-2xl mx-auto mb-12">

Here are some of my recent projects that showcase my skills and experience.

</p>

</FadeUp>

{*/\* Filter Buttons \*/*}

<FadeUp delay={0.3}>

<div className="flex flex-wrap gap-2 justify-center mb-12">

<Button

variant={filter === 'all' ? 'default' : 'outline'}

size="sm"

onClick={() => setFilter('all')}

>

All

</Button>

{technologies.map((tech) => (

<Button

key={tech}

variant={filter === tech ? 'default' : 'outline'}

size="sm"

onClick={() => setFilter(tech)}

>

{tech}

</Button>

))}

</div>

</FadeUp>

{*/\* Projects Grid \*/*}

<StaggerChildren className="grid md:grid-cols-2 lg:grid-cols-3 gap-6">

{filteredProjects.map((project) => (

<motion.div

key={project.id}

variants={{

hidden: { opacity: 0, y: 20 },

show: { opacity: 1, y: 0 }

}}

whileHover={{ y: -5 }}

className="group"

>

<Card className="h-full overflow-hidden transition-all duration-300 group-hover:shadow-xl">

<CardContent className="p-0">

{*/\* Project Image \*/*}

<div className="aspect-video bg-muted relative overflow-hidden">

<div className="w-full h-full bg-gradient-to-br from-primary/20 to-primary/10 flex items-center justify-center">

<span className="text-4xl">🚀</span>

</div>

<div className="absolute inset-0 bg-black/60 opacity-0 group-hover:opacity-100 transition-opacity duration-300 flex items-center justify-center gap-4">

<Button size="sm" className="gap-2">

<ExternalLink className="h-4 w-4" />

Live Demo

</Button>

<Button variant="outline" size="sm" className="gap-2">

<Github className="h-4 w-4" />

Code

</Button>

</div>

</div>

{*/\* Project Info \*/*}

<div className="p-6">

<h3 className="font-semibold text-lg mb-2 group-hover:text-primary transition-colors">

{project.title}

</h3>

<p className="text-muted-foreground text-sm mb-4">

{project.description}

</p>

<div className="flex flex-wrap gap-2 mb-4">

{project.technologies.map((tech) => (

<Badge key={tech} variant="secondary" className="text-xs">

{tech}

</Badge>

))}

</div>

<div className="flex gap-4">

<Button variant="ghost" size="sm" className="gap-2 text-xs">

View Details

<ArrowRight className="h-3 w-3" />

</Button>

</div>

</div>

</CardContent>

</Card>

</motion.div>

))}

</StaggerChildren>

</div>

</section>

)

}

**Commit 7: "feat: implement projects section with filtering and hover effects"**

bash

git add .

git commit -m "feat: implement projects section with filtering and hover effects

- Create projects data structure

- Build filterable projects grid

- Add hover animations with Framer Motion

- Implement technology badges

- Create project cards with overlay effects"

**Day 8: Experience Section**

**8.1 Create Experience Section** (src/components/sections/Experience.jsx)

javascript

"use client"

import { FadeUp } from '@/components/animations/FadeUp'

import { experiences } from '@/constants/data'

import { Calendar, MapPin } from 'lucide-react'

export default function Experience() {

return (

<section id="experience" className="py-20">

<div className="container">

<FadeUp>

<h2 className="text-3xl md:text-4xl font-bold text-center mb-4">Experience</h2>

</FadeUp>

<FadeUp delay={0.2}>

<p className="text-xl text-muted-foreground text-center max-w-2xl mx-auto mb-12">

My professional journey and the companies I've worked with.

</p>

</FadeUp>

{*/\* Timeline \*/*}

<div className="max-w-3xl mx-auto">

{experiences.map((experience, index) => (

<FadeUp key={experience.id} delay={index \* 0.1}>

<div className="flex gap-6 group">

{*/\* Timeline line \*/*}

<div className="flex flex-col items-center">

<div className="w-4 h-4 bg-primary rounded-full border-4 border-background group-hover:scale-125 transition-transform" />

{index !== experiences.length - 1 && (

<div className="w-0.5 h-full bg-border mt-2" />

)}

</div>

{*/\* Content \*/*}

<div className="flex-1 pb-12 group-hover:translate-x-2 transition-transform">

<div className="bg-card border rounded-lg p-6 shadow-sm group-hover:shadow-md transition-shadow">

<div className="flex flex-col sm:flex-row sm:justify-between sm:items-start gap-2 mb-4">

<div>

<h3 className="font-semibold text-lg">{experience.position}</h3>

<p className="text-primary font-medium">{experience.company}</p>

</div>

<div className="flex items-center gap-2 text-sm text-muted-foreground">

<Calendar className="h-4 w-4" />

<span>{experience.period}</span>

</div>

</div>

<p className="text-muted-foreground mb-4">

{experience.description}

</p>

<div className="flex flex-wrap gap-2">

{experience.technologies.map((tech) => (

<span

key={tech}

className="px-2 py-1 bg-primary/10 text-primary rounded text-xs font-medium"

>

{tech}

</span>

))}

</div>

</div>

</div>

</div>

</FadeUp>

))}

</div>

</div>

</section>

)

}

**Commit 8: "feat: create experience section with timeline layout"**

bash

git add .

git commit -m "feat: create experience section with timeline layout

- Build timeline component with vertical line

- Add hover animations for timeline items

- Implement experience cards with technology tags

- Create responsive layout for timeline

- Add calendar and location icons"

**Day 9: Contact Section**

**9.1 Create Contact Section** (src/components/sections/Contact.jsx)

javascript

"use client"

import { useState } from 'react'

import { FadeUp } from '@/components/animations/FadeUp'

import { Button } from '@/components/ui/button'

import { Input } from '@/components/ui/input'

import { Textarea } from '@/components/ui/textarea'

import { Card, CardContent } from '@/components/ui/card'

import { Mail, Phone, MapPin, Send } from 'lucide-react'

export default function Contact() {

const [formData, setFormData] = useState({

name: '',

email: '',

subject: '',

message: ''

})

const handleSubmit = (e) => {

e.preventDefault()

*// Handle form submission*

console.log('Form submitted:', formData)

}

const handleChange = (e) => {

setFormData(prev => ({

...prev,

[e.target.name]: e.target.value

}))

}

const contactMethods = [

{

icon: Mail,

label: 'Email',

value: 'your.email@example.com',

href: 'mailto:your.email@example.com'

},

{

icon: Phone,

label: 'Phone',

value: '+1 (555) 123-4567',

href: 'tel:+15551234567'

},

{

icon: MapPin,

label: 'Location',

value: 'Your City, Country',

href: '#'

}

]

return (

<section id="contact" className="py-20 bg-muted/50">

<div className="container">

<FadeUp>

<h2 className="text-3xl md:text-4xl font-bold text-center mb-4">Get In Touch</h2>

</FadeUp>

<FadeUp delay={0.2}>

<p className="text-xl text-muted-foreground text-center max-w-2xl mx-auto mb-12">

Have a project in mind? Let's discuss how we can work together.

</p>

</FadeUp>

<div className="grid lg:grid-cols-2 gap-12">

{*/\* Contact Information \*/*}

<div className="space-y-8">

<FadeUp>

<h3 className="text-2xl font-semibold">Let's Connect</h3>

</FadeUp>

<FadeUp delay={0.1}>

<p className="text-muted-foreground">

I'm always interested in new opportunities and collaborations.

Whether you have a project in mind or just want to say hello,

feel free to reach out!

</p>

</FadeUp>

<div className="space-y-4">

{contactMethods.map((method, index) => (

<FadeUp key={method.label} delay={0.2 + index \* 0.1}>

<Card>

<CardContent className="p-4">

<a

href={method.href}

className="flex items-center gap-4 group"

>

<div className="p-2 bg-primary/10 rounded-lg group-hover:bg-primary/20 transition-colors">

<method.icon className="h-5 w-5 text-primary" />

</div>

<div>

<p className="font-medium text-sm">{method.label}</p>

<p className="text-muted-foreground text-sm">{method.value}</p>

</div>

</a>

</CardContent>

</Card>

</FadeUp>

))}

</div>

</div>

{*/\* Contact Form \*/*}

<FadeUp delay={0.3}>

<Card>

<CardContent className="p-6">

<form onSubmit={handleSubmit} className="space-y-6">

<div className="grid sm:grid-cols-2 gap-4">

<div className="space-y-2">

<label htmlFor="name" className="text-sm font-medium">

Name

</label>

<Input

id="name"

name="name"

value={formData.name}

onChange={handleChange}

placeholder="Your name"

required

/>

</div>

<div className="space-y-2">

<label htmlFor="email" className="text-sm font-medium">

Email

</label>

<Input

id="email"

name="email"

type="email"

value={formData.email}

onChange={handleChange}

placeholder="your.email@example.com"

required

/>

</div>

</div>

<div className="space-y-2">

<label htmlFor="subject" className="text-sm font-medium">

Subject

</label>

<Input

id="subject"

name="subject"

value={formData.subject}

onChange={handleChange}

placeholder="What's this about?"

required

/>

</div>

<div className="space-y-2">

<label htmlFor="message" className="text-sm font-medium">

Message

</label>

<Textarea

id="message"

name="message"

value={formData.message}

onChange={handleChange}

placeholder="Tell me about your project..."

rows={5}

required

/>

</div>

<Button type="submit" className="w-full gap-2">

<Send className="h-4 w-4" />

Send Message

</Button>

</form>

</CardContent>

</Card>

</FadeUp>

</div>

</div>

</section>

)

}

**Commit 9: "feat: implement contact section with form and info cards"**

bash

git add .

git commit -m "feat: implement contact section with form and info cards

- Create contact form with validation

- Add contact method cards with icons

- Implement form state management

- Build responsive two-column layout

- Add hover effects and transitions"

**Phase 4: Advanced Features & Polish (Days 11-14)**

**Day 10: Main Page Integration & Smooth Scrolling**

**10.1 Update Main Page** (src/app/page.jsx)

javascript

import Header from '@/components/layout/Header'

import Footer from '@/components/layout/Footer'

import Hero from '@/components/sections/Hero'

import About from '@/components/sections/About'

import Projects from '@/components/sections/Projects'

import Experience from '@/components/sections/Experience'

import Contact from '@/components/sections/Contact'

export default function Home() {

return (

<main className="min-h-screen">

<Header />

<Hero />

<About />

<Projects />

<Experience />

<Contact />

<Footer />

</main>

)

}

**10.2 Add Smooth Scrolling** (src/components/layout/SmoothScroll.jsx)

javascript

"use client"

import { useEffect } from 'react'

import { usePathname } from 'next/navigation'

export default function SmoothScroll({ children }) {

const pathname = usePathname()

useEffect(() => {

const handleClick = (event) => {

const target = event.target.closest('a')

if (!target) return

const href = target.getAttribute('href')

if (href && href.startsWith('#')) {

event.preventDefault()

const element = document.querySelector(href)

if (element) {

element.scrollIntoView({ behavior: 'smooth' })

}

}

}

document.addEventListener('click', handleClick)

return () => document.removeEventListener('click', handleClick)

}, [pathname])

return children

}

**10.3 Update Layout with Smooth Scroll** (src/app/layout.jsx)

javascript

import { Inter } from 'next/font/google'

import './globals.css'

import { ThemeProvider } from '@/contexts/ThemeContext'

import SmoothScroll from '@/components/layout/SmoothScroll'

const inter = Inter({ subsets: ['latin'] })

export const metadata = {

title: 'Your Name - Portfolio',

description: 'Personal portfolio showcasing projects and experience',

}

export default function RootLayout({ children }) {

return (

<html lang="en" suppressHydrationWarning>

<body className={inter.className}>

<ThemeProvider>

<SmoothScroll>

{children}

</SmoothScroll>

</ThemeProvider>

</body>

</html>

)

}

**Commit 10: "feat: integrate all sections and add smooth scrolling"**

bash

git add .

git commit -m "feat: integrate all sections and add smooth scrolling

- Combine all sections in main page

- Implement smooth scroll for navigation

- Update layout with smooth scroll provider

- Ensure proper section ordering

- Test navigation functionality"

**Day 11: Performance Optimization**

**11.1 Add Loading States** (src/components/animations/PageLoader.jsx)

javascript

"use client"

import { motion } from 'framer-motion'

import { useEffect, useState } from 'react'

export default function PageLoader() {

const [isLoading, setIsLoading] = useState(true)

useEffect(() => {

const timer = setTimeout(() => setIsLoading(false), 2000)

return () => clearTimeout(timer)

}, [])

if (!isLoading) return null

return (

<motion.div

initial={{ opacity: 1 }}

exit={{ opacity: 0 }}

className="fixed inset-0 z-50 flex items-center justify-center bg-background"

>

<motion.div

animate={{

scale: [1, 1.2, 1],

rotate: [0, 180, 360],

}}

transition={{

duration: 2,

repeat: Infinity,

ease: "easeInOut",

}}

className="w-16 h-16 border-4 border-primary border-t-transparent rounded-full"

/>

</motion.div>

)

}

**11.2 Optimize Images** - Create custom image component

**Commit 11: "perf: add loading animations and optimize performance"**

bash

git add .

git commit -m "perf: add loading animations and optimize performance

- Create page loader with Framer Motion

- Add loading states for better UX

- Optimize component loading

- Improve animation performance

- Add exit animations"

**Phase 5: Final Polish & Deployment (Days 15-16)**

**Day 12: Final Testing & Content**

**12.1 Update Content** - Replace placeholder content with real data

**12.2 Test Responsive Design** - Ensure all components work on mobile/tablet

**Commit 12: "content: update with real portfolio content"**

bash

git add .

git commit -m "content: update with real portfolio content

- Replace placeholder text with personal information

- Add real project descriptions and links

- Update experience with actual work history

- Include real contact information

- Add personal bio and skills"

**Day 13: Deployment Preparation**

**13.1 Update Package.json** - Add build scripts

**13.2 Create Vercel Configuration** (vercel.json)

json

{

"version": 2,

"buildCommand": "npm run build",

"outputDirectory": ".next",

"devCommand": "npm run dev",

"installCommand": "npm install"

}

**Commit 13: "build: prepare for deployment"**

bash

git add .

git commit -m "build: prepare for deployment

- Update package.json scripts

- Add Vercel configuration

- Optimize build settings

- Test production build

- Fix any build warnings"

**Day 14: Deployment & Launch**

**14.1 Deploy to Vercel**

bash

npm run build

*# Deploy to Vercel through GitHub or Vercel CLI*

**14.2 Final Testing** - Test live site thoroughly

**Commit 14: "deploy: launch portfolio live"**

bash

git add .

git commit -m "deploy: launch portfolio live

- Successful deployment to Vercel

- Domain configuration complete

- All features working in production

- Performance optimized

- Portfolio is live and accessible"

**Final Git Log Structure:**

text

\* deploy: launch portfolio live

\* build: prepare for deployment

\* content: update with real portfolio content

\* perf: add loading animations and optimize performance

\* feat: integrate all sections and add smooth scrolling

\* feat: implement contact section with form and info cards

\* feat: create experience section with timeline layout

\* feat: implement projects section with filtering and hover effects

\* feat: build about section with skills grid and personal story

\* feat: create animated hero section with typewriter effect

\* feat: implement reusable animation components

\* feat: create layout components with responsive navigation

\* feat: implement theme system with context provider

\* feat: initial project setup with Next.js, Tailwind, and ShadCN