

# Next.js Project (Converted from Figma Template)

## File: package.json

```
{
  "name": "figma-portfolio-nextjs",
  "version": "1.0.0",
  "private": true,
  "scripts": {
    "dev": "next dev",
    "build": "next build",
    "start": "next start",
    "lint": "next lint"
  },
  "dependencies": {
    "next": "14.0.0",
    "react": "18.2.0",
    "react-dom": "18.2.0"
  },
  "devDependencies": {
    "autoprefixer": "^10.4.14",
    "postcss": "^8.4.24",
    "tailwindcss": "^3.5.2",
    "typescript": "^5.0.4"
  }
}
```

## File: next.config.js

```
/** @type {import('next').NextConfig} */
const nextConfig = {
  reactStrictMode: true,
  images: { domains: ['images.unsplash.com'] },
}
module.exports = nextConfig
```

## File: tsconfig.json

```
{
  "compilerOptions": {
    "target": "ES2022",
    "lib": [
      "DOM",
      "DOM.Iterable",
      "ES2022"
    ],
    "allowJs": true,
    "skipLibCheck": true,
    "strict": true,
    "forceConsistentCasingInFileNames": true,
    "noEmit": true,
    "esModuleInterop": true,
    "module": "ESNext",
    "moduleResolution": "Node",
    "resolveJsonModule": true,
    "isolatedModules": true,
    "jsx": "preserve",
    "incremental": true
  },
  "include": [
    "next-env.d.ts",
    "**/*.ts",
    "**/*.tsx"
  ]
}
```

```

    ],
    "exclude": [
      "node_modules"
    ]
  }
}

```

## File: next-env.d.ts

```

/// <reference types="next" />
/// <reference types="next/types/global" />

```

## File: tailwind.config.js

```

module.exports = {
  content: [
    './app/**/*.ts,tsx',
    './components/**/*.ts,tsx',
  ],
  theme: {
    extend: {
      colors: {
        primary: '#0f172a',
        accent: '#06b6d4'
      }
    },
  },
  plugins: [],
}

```

## File: postcss.config.js

```

module.exports = {
  plugins: {
    tailwindcss: {},
    autoprefixer: {},
  },
}

```

## File: README.md

```

# Generated Next.js Portfolio (from uploaded template)

This project was generated automatically from the uploaded template zip.

## What is included
- Next.js with App Router (TypeScript)
- Tailwind CSS config files
- Components: Header, Hero, Projects, Footer
- Public assets copied from the uploaded template (in `public/`)
- Basic responsive layout

## How to run
1. `cd` into the project folder.
2. Install: `npm install`
3. Run dev: `npm run dev`

Customize the pages/components/styles to match the Figma layout further.

```

## File: app/layout.tsx

```
import './globals.css'
import type { ReactNode } from 'react'

export const metadata = {
  title: 'Designer Developer – Portfolio',
  description: 'Portfolio generated from a Figma template. Responsive, accessible and production-ready',
  openGraph: {
    title: 'Designer Developer – Portfolio',
    description: 'Portfolio generated from a Figma template.',
    images: ['/og-image.png'],
  },
}

export default function RootLayout({ children }: { children: ReactNode }) {
  return (
    <html lang="en">
      <head>
        <link rel="preconnect" href="https://fonts.googleapis.com"/>
        <link rel="preconnect" href="https://fonts.gstatic.com" crossOrigin="true"/>
        <link href="https://fonts.googleapis.com/css2?family=Inter:wght@300;400;600;800&display=swap" rel="stylesheet"/>
      </head>
      <body>
        <div className="min-h-screen flex flex-col bg-white text-slate-900">
          {children}
        </div>
      </body>
    </html>
  )
}
```

## File: app/page.tsx

```
import Image from 'next/image'
import Header from '../components/Header'
import Hero from '../components/Hero'
import Projects from '../components/Projects'
import Footer from '../components/Footer'

export default function Home() {
  return (
    <>
      <Header />
      <main className="flex-1">
        <Hero />
        <Projects />
      </main>
      <Footer />
    </>
  )
}
```

## File: app/api/contact/route.ts

```
import { NextResponse } from 'next/server'
import fs from 'fs/promises'
import path from 'path'

export async function POST(req: Request) {
  try {
    const body = await req.json()
    const dataDir = path.join(process.cwd(), 'data')
    await fs.mkdir(dataDir, { recursive: true })
    const file = path.join(dataDir, 'contacts.json')
    let arr = []
    try {
      const existing = await fs.readFile(file, 'utf8')
```

```

    arr = JSON.parse(existing)
  } catch (err) {
    arr = []
  }
  const entry = { id: Date.now(), ...body, createdAt: new Date().toISOString() }
  arr.push(entry)
  await fs.writeFile(file, JSON.stringify(arr, null, 2), 'utf8')
  return NextResponse.json({ ok: true })
} catch (err) {
  return NextResponse.json({ ok: false }, { status: 500 })
}
}
}

```

## File: app/about/page.tsx

```

import Image from 'next/image'

export default function About() {
  return (
    <section id="about" className="py-16">
      <div className="container grid grid-cols-1 md:grid-cols-3 gap-8 items-start">
        <div className="md:col-span-1">
          <div className="w-full max-w-sm mx-auto md:mx-0">
            <div className="relative w-64 h-64 mx-auto md:mx-0">
              <Image src="/Me.png" alt="Portrait" fill style={{objectFit:'contain'}} />
            </div>
          </div>
        </div>
        <div className="md:col-span-2">
          <h1 className="text-3xl md:text-4xl font-extrabold">About Me</h1>
          <p className="mt-4 text-lg text-slate-700">
            I'm a designer-developer who loves turning ideas into delightful products. I focus on pi
          </p>
          <h2 className="mt-6 text-xl font-semibold">Skills</h2>
          <ul className="mt-3 grid grid-cols-2 gap-2 text-sm">
            <li>React / Next.js</li>
            <li>TypeScript</li>
            <li>Tailwind CSS</li>
            <li>Design Systems</li>
            <li>Figma / Prototyping</li>
            <li>Accessibility (ally)</li>
          </ul>
        </div>
      </div>
    </section>
  )
}

```

## File: app/contact/page.tsx

```

'use client'
import { useState } from 'react'

export default function Contact() {
  const [form, setForm] = useState({name:'', email:'', message:''})
  const [status, setStatus] = useState<'idle' | 'loading' | 'success' | 'error'>('idle')

  async function handleSubmit(e: React.FormEvent) {
    e.preventDefault()
    setStatus('loading')
    try {
      const res = await fetch('/api/contact', {
        method: 'POST',
        headers: { 'Content-Type': 'application/json' },
        body: JSON.stringify(form)
      })
    }
    if (res.ok) {
      setStatus('success')
      setForm({name:'', email:'', message:''})
    }
  }
}

```

```

    } else {
      setStatus('error')
    }
  } catch (err) {
    setStatus('error')
  }
}

return (
  <section id="contact" className="py-16">
    <div className="container max-w-xl mx-auto">
      <h2 className="text-2xl font-bold">Contact</h2>
      <p className="mt-2 text-slate-600">Want to work together? Send a message.</p>

      <form onSubmit={handleSubmit} className="mt-6 grid gap-4">
        <label className="flex flex-col">
          <span className="text-sm font-medium">Name</span>
          <input required value={form.name} onChange={(e)=>setForm({...form, name:e.target.value})
            className="mt-1 p-3 rounded border focus:outline-none focus:ring-2" />
        </label>

        <label className="flex flex-col">
          <span className="text-sm font-medium">Email</span>
          <input required type="email" value={form.email} onChange={(e)=>setForm({...form, email:e
            className="mt-1 p-3 rounded border focus:outline-none focus:ring-2" />
        </label>

        <label className="flex flex-col">
          <span className="text-sm font-medium">Message</span>
          <textarea required value={form.message} onChange={(e)=>setForm({...form, message:e.target
            className="mt-1 p-3 rounded border focus:outline-none focus:ring-2 min-h-[120px]" />
        </label>

        <div className="flex items-center gap-4">
          <button disabled={status==='loading'} type="submit" className="px-4 py-2 rounded bg-acce
            {status==='loading' ? 'Sending...' : 'Send Message'}
          </button>
          {status==='success' && <div className="text-green-600">Message sent – thank you!</div>}
          {status==='error' && <div className="text-red-600">Something went wrong. Try again.</div>}
        </div>
      </form>
    </div>
  </section>
)
}

```

## File: components/Header.tsx

```

'use client'
import Link from 'next/link'
import { useState, useEffect } from 'react'

export default function Header(){
  const [open, setOpen] = useState(false)

  // close on resize to bigger screens
  useEffect(()=>{
    function onResize(){ if (window.innerWidth >= 768) setOpen(false) }
    window.addEventListener('resize', onResize)
    return ()=>window.removeEventListener('resize', onResize)
  },[])

  return (
    <header className="bg-white shadow-sm sticky top-0 z-40">
      <div className="container flex items-center justify-between py-4">
        <Link href="/" className="text-xl font-bold" aria-label="Home">Designer Developer</Link>

        <nav className="hidden md:flex gap-6" aria-label="Primary">
          <a href="#about" className="hover:underline">About</a>
          <a href="#projects" className="hover:underline">Projects</a>
          <a href="#contact" className="hover:underline">Contact</a>
        </nav>
      <div className="md:hidden">

```

```

        <button aria-expanded={open} aria-label="Toggle menu" onClick={()=>setOpen(!open)}
          className="p-2 rounded focus:outline-none focus:ring-2">
          <span>{open ? 'X' : '■'}</span>
        </button>
      </div>
    </div>

    { /* Mobile menu */ }
    <div className={`md:hidden transition-max-h duration-300 overflow-hidden ${open ? 'max-h-40' : ''}`>
      <nav className="px-4 pb-4 flex flex-col gap-2" aria-label="Mobile">
        <a href="#about" onClick={()=>setOpen(false)} className="block py-2">About</a>
        <a href="#projects" onClick={()=>setOpen(false)} className="block py-2">Projects</a>
        <a href="#contact" onClick={()=>setOpen(false)} className="block py-2">Contact</a>
      </nav>
    </div>
  </header>
)
}

```

## File: components/Hero.tsx

```

import Image from 'next/image'

export default function Hero(){
  return (
    <section aria-label="Introduction" className="py-12">
      <div className="container grid grid-cols-1 md:grid-cols-2 gap-8 items-center">
        <div className="animate-fade-up">
          <h1 className="text-4xl md:text-5xl font-extrabold">Hi, I'm Designer Developer</h1>
          <p className="mt-4 text-lg text-slate-700">I build beautiful interfaces and delightful user experiences</p>
          <div className="mt-6 flex gap-4">
            <a href="#projects" className="px-4 py-2 rounded bg-accent text-white">View Projects</a>
            <a href="#contact" className="px-4 py-2 rounded border">Contact</a>
          </div>
        </div>
        <div className="flex justify-center md:justify-end animate-fade-up">
          <div className="w-64 h-64 relative">
            <Image src="/Me.png" alt="Me" fill style={{objectFit:'contain'}} />
          </div>
        </div>
      </div>
    </section>
  )
}

```

## File: components/Projects.tsx

```

import Image from 'next/image'

export default function Projects(){
  const projects = [
    {title:'Project One', img:'/Portfolio.png', desc:'A short description.'},
    {title:'Project Two', img:'/Portfolio-1.png', desc:'A short description.'},
    {title:'Project Three', img:'/Portfolio-2.png', desc:'A short description.'},
  ]
  return (
    <section id="projects" className="py-12 bg-slate-50">
      <div className="container">
        <h2 className="text-2xl font-bold animate-fade-up">Selected Projects</h2>
        <div className="mt-6 grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-3 gap-6">
          {projects.map((p)=>(
            <article key={p.title} className="rounded-lg overflow-hidden shadow-sm bg-white" aria-label={p.title}>
              <div className="relative h-48 w-full">
                <Image src={p.img} alt={p.title} fill style={{objectFit:'cover'}} />
              </div>
              <div className="p-4">
                <h3 id={p.title} className="font-semibold">{p.title}</h3>
                <p className="mt-2 text-sm text-slate-600">{p.desc}</p>
              </div>
            </article>
          ))}
        </div>
      </div>
    </section>
  )
}

```

```

        </article>
      )})
    </div>
  </div>
</section>
)
}

```

## File: components/Footer.tsx

```

export default function Footer(){
  return (
    <footer className="border-t mt-12" role="contentinfo">
      <div className="container py-6 text-center text-sm text-slate-600">
        © {new Date().getFullYear()} Your Name. – Built from a Figma template.
      </div>
    </footer>
  )
}

```

## File: styles/globals.css

```

@tailwind base;
@tailwind components;
@tailwind utilities;

/* Design tokens approximated from template */
:root{
  --color-bg: #ffffff;
  --color-foreground: #0f172a;
  --color-accent: #06b6d4;
  --ff-sans: Inter, ui-sans-serif, system-ui, -apple-system, 'Segoe UI', Roboto, 'Helvetica Neue', A
}

html, body, #__next { height: 100%; }
body {
  @apply bg-[color:var(--color-bg)] text-[color:var(--color-foreground)] antialiased;
  font-family: var(--ff-sans);
}

/* Container */
.container {
  max-width: 1100px;
  margin-left: auto;
  margin-right: auto;
  padding-left: 1rem;
  padding-right: 1rem;
}

/* Accent utilities */
.bg-accent { background-color: var(--color-accent); }
.text-accent { color: var(--color-accent); }

/* Simple fade-up animation used for hero and projects */
@keyframes fadeUp {
  from { opacity: 0; transform: translateY(8px); }
  to { opacity: 1; transform: translateY(0); }
}
.animate-fade-up { animation: fadeUp 600ms cubic-bezier(.2,.9,.3,1) both; }

/* Accessibility: focus ring */
:focus { outline: none; box-shadow: 0 0 0 3px rgba(6,182,212,0.15); border-radius: 6px; }

/* Mobile menu transition helper */
.transition-max-h { transition-property: max-height; transition-timing-function: cubic-bezier(.2,.9,

```