

Next.js 企业官网完整源码模板（基于 Wagtail Headless CMS）

1. 技术栈

Frontend

```
Next.js 15  
React 19  
TypeScript  
TailwindCSS  
Shadcn UI  
Framer Motion  
React Query  
Zustand  
next-intl
```

2. 项目目录结构

```
frontend/  
├── app/  
├── blocks/  
├── components/  
├── services/  
├── hooks/  
├── stores/  
├── providers/  
├── lib/  
├── styles/  
├── types/  
├── messages/  
├── public/  
├── middleware.ts  
├── next.config.ts  
├── tailwind.config.ts  
├── package.json  
└── tsconfig.json
```

3. package.json

```
{
  "name": "enterprise-website",
  "version": "1.0.0",
  "private": true,
  "scripts": {
    "dev": "next dev",
    "build": "next build",
    "start": "next start",
    "lint": "next lint"
  },
  "dependencies": {
    "next": "15.0.0",
    "react": "19.0.0",
    "react-dom": "19.0.0",
    "tailwindcss": "^3.4.0",
    "framer-motion": "^11.0.0",
    "zustand": "^4.5.0",
    "@tanstack/react-query": "^5.0.0",
    "next-intl": "^3.0.0",
    "lucide-react": "^0.400.0"
  }
}
```

4. Next.js 配置

next.config.ts

```
import type { NextConfig } from 'next'

const nextConfig: NextConfig = {
  reactStrictMode: true,
  images: {
    remotePatterns: [
      {
        protocol: 'https',
        hostname: '***',
      },
    ],
  },
}

export default nextConfig
```

5. Tailwind 配置

tailwind.config.ts

```
import type { Config } from 'tailwindcss'

const config: Config = {
  content: [
    './app/**/*.js,ts,jsx,tsx',
    './components/**/*.js,ts,jsx,tsx',
    './blocks/**/*.js,ts,jsx,tsx',
  ],
  theme: {
    extend: {
      colors: {
        primary: '#2563EB',
        dark: '#0F172A',
      },
    },
  },
  plugins: [],
}

export default config
```

6. App Router 架构

```
app/
├── layout.tsx
├── page.tsx
├── globals.css
├── products/
├── solutions/
├── blog/
├── about/
├── careers/
└── contact/
```

7. 全局 Layout

app/layout.tsx

```
import './globals.css'
import Header from '@components/layout/Header'
import Footer from '@components/layout/Footer'

export default function RootLayout({
  children,
}: {
  children: React.ReactNode
}) {
  return (
    <html lang="zh">
      <body className="bg-white text-slate-900">
        <Header />

        <main>{children}</main>

        <Footer />
      </body>
    </html>
  )
}
```

8. 首页入口

app/page.tsx

```
import Hero from '@blocks/hero/Hero'
import Stats from '@blocks/stats/Stats'
import Products from '@blocks/products/Products'
import Solutions from '@blocks/solutions/Solutions'
import Cases from '@blocks/cases/Cases'
import CTA from '@blocks/cta/CTA'

export default function HomePage() {
  return (
    <>
      <Hero />
      <Stats />
      <Products />
      <Solutions />
      <Cases />
    </>
  )
}
```

```
    <CTA />
  </>
)
}
```

9. Header 组件

components/layout/Header.tsx

```
'use client'

import Link from 'next/link'

const navItems = [
  {
    title: '产品',
    href: '/products',
  },
  {
    title: '解决方案',
    href: '/solutions',
  },
  {
    title: '案例',
    href: '/cases',
  },
  {
    title: '博客',
    href: '/blog',
  },
]

export default function Header() {
  return (
    <header className="sticky top-0 z-50 border-b bg-white/80 backdrop-blur">
      <div className="mx-auto flex h-16 max-w-7xl items-center justify-between px-6">
        <Link href="/" className="text-2xl font-bold text-primary">
          TechFlow
        </Link>

        <nav className="hidden gap-8 md:flex">
          {navItems.map((item) => (
            <Link key={item.href} href={item.href}>
              {item.title}
            </Link>
          ))}
        </nav>
      </div>
    </header>
  )
}
```

```

    <button className="rounded-xl bg-primary px-5 py-2 text-white">
      联系我们
    </button>
  </div>
</header>
)
}

```

10. Hero Block

blocks/hero/Hero.tsx

```

import { ArrowRight } from 'lucide-react'

export default function Hero() {
  return (
    <section className="relative overflow-hidden bg-gradient-to-b from-slate-50 to-white py-32">
      <div className="mx-auto grid max-w-7xl grid-cols-1 gap-12 px-6 lg:grid-cols-2">
        <div className="flex flex-col justify-center">
          <div className="mb-6 inline-flex w-fit rounded-full border border-blue-200 bg-blue-50 px-4 py-2 text-sm text-primary">
            AI Native ☐ Cloud Native ☐ Enterprise
          </div>

          <h1 className="mb-6 text-6xl font-bold leading-tight text-slate-900">
            用技术创新
            <br />
            驱动企业未来
          </h1>

          <p className="mb-10 text-lg leading-8 text-slate-600">
            构建企业级 AI 与云原生平台 ☐ 帮助企业实现数字化升级 ☐
          </p>

          <div className="flex gap-4">
            <button className="flex items-center gap-2 rounded-2xl bg-primary px-8 py-4 text-white">
              预约演示
              <ArrowRight size={18} />
            </button>

            <button className="rounded-2xl border border-slate-200 px-8 py-4">
              查看产品
            </button>
          </div>
        </div>
      </div>
    </section>
  )
}

```

```

    <div className="relative flex items-center justify-center">
      <div className="h-[500px] w-full rounded-[32px] bg-gradient-to-br from-blue-500 to-cyan-400
shadow-2xl" />
    </div>
  </div>
</section>
)
}

```

11. Stats Block

blocks/stats/Stats.tsx

```

const stats = [
  {
    label: '企业客户',
    value: '500+',
  },
  {
    label: '技术专家',
    value: '200+',
  },
  {
    label: '服务可用性',
    value: '99.9%',
  },
  {
    label: '行业方案',
    value: '20+',
  },
]

export default function Stats() {
  return (
    <section className="py-20">
      <div className="mx-auto grid max-w-6xl grid-cols-2 gap-8 px-6 lg:grid-cols-4">
        {stats.map((item) => (
          <div
            key={item.label}
            className="rounded-3xl border border-slate-100 bg-white p-10 text-center shadow-sm"
          >
            <div className="mb-3 text-5xl font-bold text-primary">
              {item.value}
            </div>

            <div className="text-slate-600">{item.label}</div>
          </div>
        ))}
      </div>
    </section>
  )
}

```

```

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

```

12. Products Block

blocks/products/Products.tsx

```

const products = [
  {
    title: 'AI Platform',
    description: '企业级 AI 开发平台',
  },
  {
    title: 'DataInsight',
    description: '数据分析与治理平台',
  },
  {
    title: 'IoT Hub',
    description: '物联网数据平台',
  },
]

export default function Products() {
  return (
    <section className="bg-slate-50 py-24">
      <div className="mx-auto max-w-7xl px-6">
        <div className="mb-16">
          <h2 className="mb-4 text-5xl font-bold">
            热门产品与解决方案
          </h2>

          <p className="text-slate-600">
            企业级数字化平台产品矩阵
          </p>
        </div>

        <div className="grid gap-8 lg:grid-cols-3">
          {products.map((item) => (
            <div
              key={item.title}
              className="rounded-[32px] bg-white p-8 shadow-sm transition hover:translate-y-2
                hover:shadow-xl"
            >
              <div className="mb-6 h-56 rounded-2xl bg-gradient-to-br from-blue-100 to-cyan-100" />

```



```

    <h3 className="mb-3 text-2xl font-semibold">
      {item.title}
    </h3>

    <p className="text-slate-600">
      {item.description}
    </p>
  </div>
  )}
</div>
</div>
</section>
)
}

```

13. CTA Block

blocks/cta/CTA.tsx

```

export default function CTA() {
  return (
    <section className="py-32">
      <div className="mx-auto max-w-6xl px-6">
        <div className="rounded-[40px] bg-gradient-to-r from-blue-600 to-cyan-500 p-20 text-center text-white">
          <h2 className="mb-6 text-5xl font-bold">
            开启企业数字化升级
          </h2>

          <p className="mb-10 text-xl text-blue-100">
            与amp;我们一起构建 AI Native 企业平台
          </p>

          <button className="rounded-2xl bg-white px-10 py-4 text-lg font-semibold text-primary">
            联系销售
          </button>
        </div>
      </div>
    </section>
  )
}

```

14. Footer 组件

components/layout/Footer.tsx

```
export default function Footer() {
  return (
    <footer className="bg-slate-950 py-20 text-slate-400">
      <div className="mx-auto grid max-w-7xl gap-12 px-6 lg:grid-cols-4">
        <div>
          <div className="mb-6 text-3xl font-bold text-white">
            TechFlow
          </div>

          <p>
            企业级 AI 与数字化平台解决方案提供商。
          </p>
        </div>

        <div>
          <h4 className="mb-6 text-lg font-semibold text-white">
            产品
          </h4>

          <ul className="space-y-3">
            <li>AI Platform</li>
            <li>DataInsight</li>
            <li>IoT Hub</li>
          </ul>
        </div>

        <div>
          <h4 className="mb-6 text-lg font-semibold text-white">
            解决方案
          </h4>

          <ul className="space-y-3">
            <li>智慧能源</li>
            <li>智慧工厂</li>
            <li>智慧物流</li>
          </ul>
        </div>

        <div>
          <h4 className="mb-6 text-lg font-semibold text-white">
            联系我们
          </h4>

          <ul className="space-y-3">
            <li>contact@techflow.com</li>
          </ul>
        </div>
      </div>
    </footer>
  )
}
```

```
        <li>400-888-8888</li>
      </ul>
    </div>
  </div>
</footer>
)
}
```

15. API Client

services/api-client.ts

```
export async function fetchAPI(url: string) {
  const response = await fetch(
    `${process.env.NEXT_PUBLIC_API_URL}${url}`,
    {
      next: {
        revalidate: 60,
      },
    }
  )

  if (!response.ok) {
    throw new Error('API Error')
  }

  return response.json()
}
```

16. Blog Service

services/blog.service.ts

```
import { fetchAPI } from './api-client'

export async function getBlogs() {
  return fetchAPI('/api/v1/blogs/')
}
```

17. Zustand Store

stores/app.store.ts

```
import { create } from 'zustand'

interface AppStore {
  theme: string
  setTheme: (theme: string) => void
}

export const useAppStore = create<AppStore>((set) => ({
  theme: 'light',
  setTheme: (theme) => set({ theme }),
}))
```

18. React Query Provider

providers/react-query-provider.tsx

```
'use client'

import {
  QueryClient,
  QueryClientProvider,
} from '@tanstack/react-query'

const queryClient = new QueryClient()

export default function ReactQueryProvider({
  children,
}: {
  children: React.ReactNode
}) {
  return (
    <QueryClientProvider client={queryClient}>
      {children}
    </QueryClientProvider>
  )
}
```

19. SEO Metadata

app/metadata.ts

```
export const metadata = {  
  title: 'TechFlow AI Platform',  
  description: '企业级 AI 与数字化平台',  
}
```

20. 多语言配置

messages/zh.json

```
{  
  "home": {  
    "title": "用技术创新驱动企业未来"  
  }  
}
```

21. Dockerfile

Dockerfile

```
FROM node:22-alpine  
  
WORKDIR /app  
  
COPY package*.json ./  
  
RUN npm install  
  
COPY . .  
  
RUN npm run build  
  
EXPOSE 3000  
  
CMD ["npm", "start"]
```

22. Docker Compose

docker-compose.yml

```
version: '3.9'

services:
  frontend:
    build: .
    ports:
      - '3000:3000'
    environment:
      NEXT_PUBLIC_API_URL: http://backend:8000
```

23. 环境变量

.env.local

```
NEXT_PUBLIC_API_URL=http://localhost:8000
NEXT_PUBLIC_SITE_URL=http://localhost:3000
```

24. 推荐扩展模块

后续建议增加

```
Blog System
Search Engine
AI Search
Dark Mode
CMS Preview
Analytics
Authentication
Multi-language
SaaS Portal
Customer Center
```

25. 推荐部署架构

```
graph TD; A[Cloudflare CDN] --> B[Nginx]; B --> C[Next.js App]; C --> D[Wagtail API]; D --> E[PostgreSQL + Redis];
```

26. 推荐企业级优化

Frontend

SSR
ISR
Image Optimization
Code Splitting
Edge Cache

Backend

Redis Cache
OpenSearch
Celery
Async Task
Webhook

27. 推荐 UI 风格

Stripe
Vercel
Linear
OpenAI
Cloudflare

28. 推荐下一步开发

下一步建议继续实现：

1. Wagtail CMS Backend 完整源码
2. StreamField 动态渲染引擎
3. 产品详情页模板
4. 博客系统
5. 搜索系统
6. AI 搜索
7. 企业级 Design System
8. SaaS 登录中心
9. Kubernetes 部署 YAML
10. Terraform 云部署
11. OpenSearch 集成
12. 企业级权限系统
13. GraphQL API
14. CMS Preview 实时预览
15. 多租户 SaaS 架构