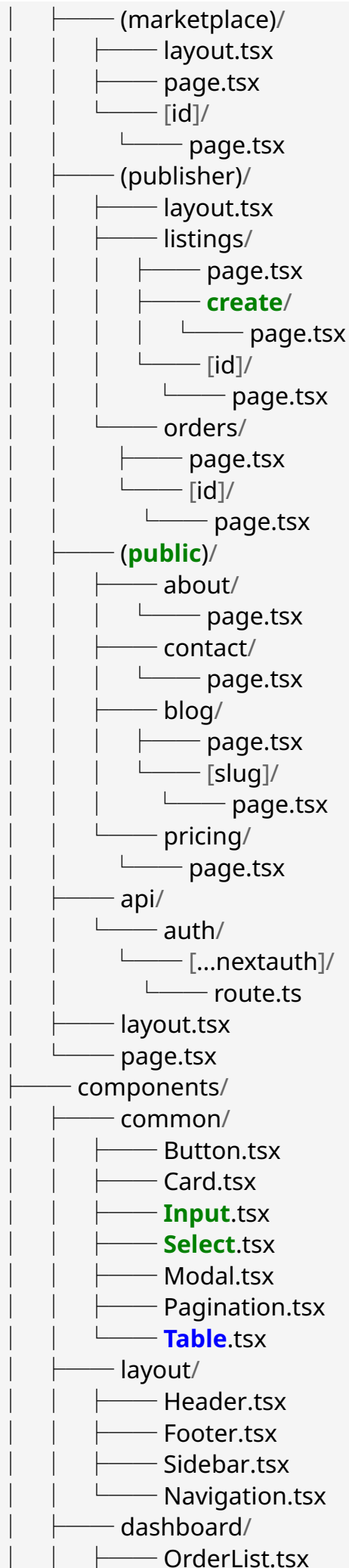


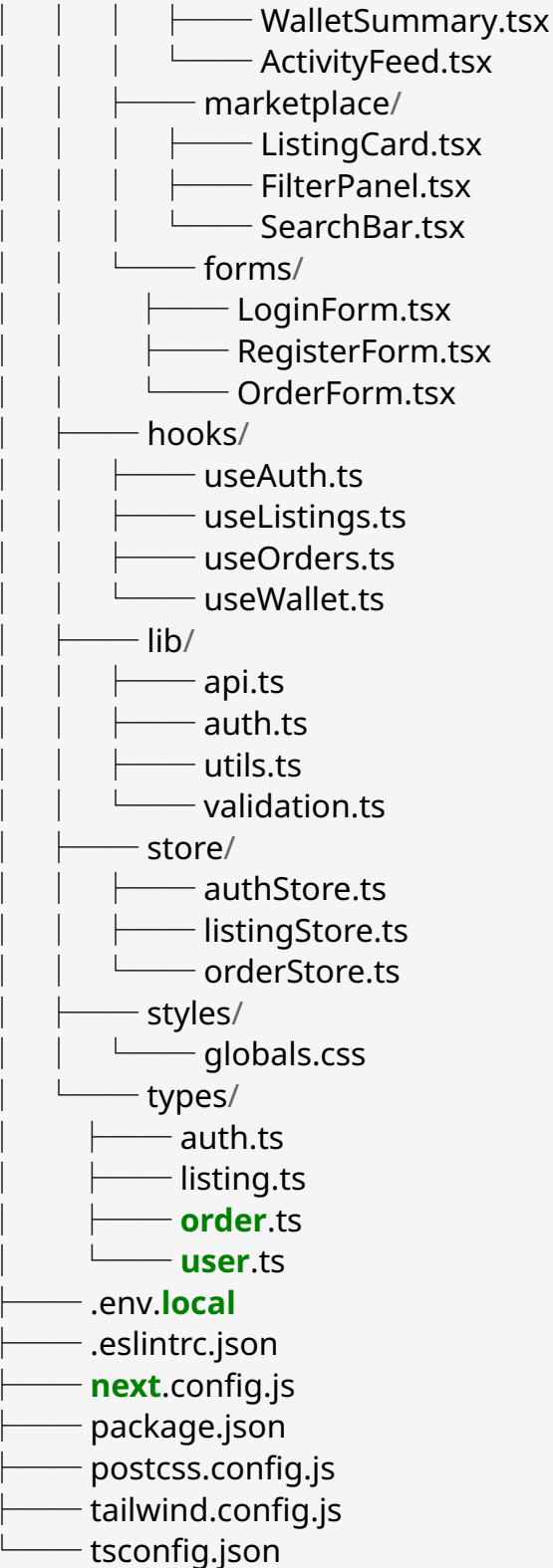
# Anic Digital Link Building Marketplace - Frontend Setup

This document provides the basic frontend setup and structure for the Anic Digital Link Building Marketplace platform using Next.js.

## Project Structure

```
anic-digital-frontend/
├── public/
│   ├── favicon.ico
│   ├── logo.svg
│   └── images/
├── src/
│   ├── app/
│   │   ├── (auth)/
│   │   │   ├── login/
│   │   │   │   └── page.tsx
│   │   │   ├── register/
│   │   │   │   └── page.tsx
│   │   │   ├── forgot-password/
│   │   │   │   └── page.tsx
│   │   │   └── reset-password/
│   │   │       └── page.tsx
│   │   ├── (dashboard)/
│   │   │   ├── layout.tsx
│   │   │   ├── page.tsx
│   │   │   ├── orders/
│   │   │   │   ├── page.tsx
│   │   │   │   └── [id]/
│   │   │   │       └── page.tsx
│   │   │   ├── wallet/
│   │   │   │   ├── page.tsx
│   │   │   │   ├── transactions/
│   │   │   │   │   └── page.tsx
│   │   │   │   ├── topup/
│   │   │   │   │   └── page.tsx
│   │   │   │   └── settings/
│   │   │   │       └── page.tsx
│   │   │   └── tools/
│   │   │       ├── page.tsx
│   │   │       ├── competitor-analysis/
│   │   │       │   └── page.tsx
```





## Initial Setup Steps

## 1. Create Next.js Project

```
npx create-next-app@latest anic-digital-frontend --typescript --tailwind --eslint --app
cd anic-digital-frontend
```

## 2. Install Dependencies

```
npm install @tanstack/react-query zustand axios react-hook-form zod @hookform/resolvers next-auth @radix-ui/react-dialog @radix-ui/react-dropdown-menu @radix-ui/react-select lucide-react recharts
```

## 3. Configure Environment Variables

Create a `.env.local` file:

```
NEXT_PUBLIC_API_URL=http://localhost:4000/api
NEXTAUTH_URL=http://localhost:3000
NEXTAUTH_SECRET=your-nextauth-secret
```

# Core Components Implementation

## Layout Components

### Root Layout (src/app/layout.tsx)

```
import './globals.css';
import { Inter } from 'next/font/google';
import { Providers } from './providers';

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

export const metadata = {
  title: 'Anic Digital - Link Building Marketplace',
  description: 'Find high-quality link building opportunities for your website',
};

export default function RootLayout({
  children,
}: {
  children: React.ReactNode;
}) {
  return (
    <html lang="en">
      <body className={inter.className}>
        <Providers>{children}</Providers>
      </body>
    </html>
  );
}
```

## Providers (src/app/providers.tsx)

```
'use client';

import { QueryClient, QueryClientProvider } from '@tanstack/react-query';
import { SessionProvider } from 'next-auth/react';
import { useState } from 'react';

export function Providers({ children }: { children: React.ReactNode }) {
  const [queryClient] = useState(() => new QueryClient({
    defaultOptions: {
      queries: {
        refetchOnWindowFocus: false,
        staleTime: 5 * 60 * 1000,
      },
    },
  }));

  return (
    <SessionProvider>
      <QueryClientProvider client={queryClient}>
        {children}
      </QueryClientProvider>
    </SessionProvider>
  );
}
```

## Header Component (src/components/layout/Header.tsx)

```
'use client';

import Link from 'next/link';
import Image from 'next/image';
import { usePathname } from 'next/navigation';
import { useSession, signOut } from 'next-auth/react';
import { useState } from 'react';
import { Menu, X, ChevronDown } from 'lucide-react';

export default function Header() {
  const { data: session } = useSession();
  const pathname = usePathname();
  const [mobileMenuOpen, setMobileMenuOpen] = useState(false);

  const navigation = [
    { name: 'Home', href: '/' },
    { name: 'Marketplace', href: '/marketplace' },
    { name: 'Pricing', href: '/pricing' },
    { name: 'About', href: '/about' },
    { name: 'Blog', href: '/blog' },
  ];
```

```

    { name: 'Contact', href: '/contact' },
  ];

  return (
    <header className="bg-white shadow-sm">
      <nav className="mx-auto flex max-w-7xl items-center justify-between p-6 lg:px-8" aria-label="Global">
        <div className="flex lg:flex-1">
          <Link href="/" className="-m-1.5 p-1.5">
            <span className="sr-only">Anic Digital</span>
            <Image
              src="/logo.svg"
              alt="Anic Digital"
              width={150}
              height={40}
              priority
            />
          </Link>
        </div>
        <div className="flex lg:hidden">
          <button
            type="button"
            className="-m-2.5 inline-flex items-center justify-center rounded-md p-2.5 text-gray-700"
            onClick={() => setMobileMenuOpen(true)}
          >
            <span className="sr-only">Open main menu</span>
            <Menu className="h-6 w-6" aria-hidden="true" />
          </button>
        </div>
        <div className="hidden lg:flex lg:gap-x-12">
          {navigation.map((item) => (
            <Link
              key={item.name}
              href={item.href}
              className={`text-sm font-semibold leading-6 ${
                pathname === item.href
                  ? 'text-indigo-600'
                  : 'text-gray-900 hover:text-indigo-600'
              }`}
            >
              {item.name}
            </Link>
          ))}
        </div>
        <div className="hidden lg:flex lg:flex-1 lg:justify-end">
          {session ? (
            <div className="flex items-center">
              <Link
                href="/dashboard"
                className="text-sm font-semibold leading-6 text-gray-900 mr-4 hover:text-indigo-600"

```

```

    >
    Dashboard
  </Link>
  <button
    onClick={() => signOut()}
    className="text-sm font-semibold leading-6 text-gray-900 hover:text-indigo-600"
  >
    Log out <span aria-hidden="true">&rarr;</span>
  </button>
</div>
):(
  <Link
    href="/login"
    className="text-sm font-semibold leading-6 text-gray-900 hover:text-indigo-600"
  >
    Log in <span aria-hidden="true">&rarr;</span>
  </Link>
)}
</div>
</nav>

{/* Mobile menu */}
{mobileMenuOpen && (
  <div className="lg:hidden">
    <div className="fixed inset-0 z-50" />
    <div className="fixed inset-y-0 right-0 z-50 w-full overflow-y-auto bg-white px-6 py-6 sm:max-w-sm sm:ring-1 sm:ring-gray-900/10">
      <div className="flex items-center justify-between">
        <Link href="/" className="-m-1.5 p-1.5">
          <span className="sr-only">Anic Digital</span>
          <Image
            src="/logo.svg"
            alt="Anic Digital"
            width={120}
            height={32}
          />
        </Link>
        <button
          type="button"
          className="-m-2.5 rounded-md p-2.5 text-gray-700"
          onClick={() => setMobileMenuOpen(false)}
        >
          <span className="sr-only">Close menu</span>
          <X className="h-6 w-6" aria-hidden="true" />
        </button>
      </div>
      <div className="mt-6 flow-root">
        <div className="-my-6 divide-y divide-gray-500/10">
          <div className="space-y-2 py-6">
            {navigation.map((item) => (

```





## Footer Component (src/components/layout/Footer.tsx)

```
import Link from 'next/link';
import Image from 'next/image';

export default function Footer() {
  return (
    <footer className="bg-gray-900 text-white">
      <div className="mx-auto max-w-7xl px-6 py-12 md:flex md:items-center md:justify-between lg:px-8">
        <div className="flex justify-center space-x-6 md:order-2">
          <Link href="#" className="text-gray-400 hover:text-gray-300">
            <span className="sr-only">Facebook</span>
            <svg className="h-6 w-6" fill="currentColor" viewBox="0 0 24 24" aria-hidden="true">
              <path fillRule="evenodd" d="M22 12c0-5.523-4.477-10-10-10S2 6.477 2 12c0 4.991 3.657 9.128 8.438 9.878v-6.987h-2.54V12h2.54V9.797c0-2.506 1.492-3.89 3.777-3.89 1.094 0 2.238.195 2.238.195v2.46h-1.26c-1.243 0-1.63.771-1.63 1.562V12h2.773l-.443 2.89h-2.33v6.988C18.343 21.128 22 16.991 22 12z" clipRule="evenodd" />
            </svg>
          </Link>
          <Link href="#" className="text-gray-400 hover:text-gray-300">
            <span className="sr-only">Twitter</span>
            <svg className="h-6 w-6" fill="currentColor" viewBox="0 0 24 24" aria-hidden="true">
              <path d="M8.29 20.251c7.547 0 11.675-6.253 11.675-11.675 0-.178 0-.355-.012-.53A8.348 8.348 0 022 5.92a8.19 8.19 0 01-2.357.646 4.118 4.118 0 00-1.804-2.27 8.224 8.224 0 01-2.605.996 4.107 4.107 0 00-6.993 3.743 11.65 11.65 0 01-8.457-4.287 4.106 4.106 0 00-1.27 5.477A4.072 4.072 0 012.8 9.713v.052a4.105 4.105 0 003.292 4.022 4.095 4.095 0 01-1.853.07 4.108 4.108 0 003.834 2.85A8.233 8.233 0 012 18.407a11.616 11.616 0 006.29 1.84" />
            </svg>
          </Link>
          <Link href="#" className="text-gray-400 hover:text-gray-300">
            <span className="sr-only">LinkedIn</span>
            <svg className="h-6 w-6" fill="currentColor" viewBox="0 0 24 24" aria-hidden="true">
              <path fillRule="evenodd" d="M19 0h-14c-2.761 0-5 2.239-5 5v14c0 2.761 2.239 5 5 5h14c2.761 0 5-2.239 5-5v-14c0-2.761-2.239-5-5-5zm-11 19h-3v-11h3v11zm-1.5-12.268c-.966 0-1.75-.79-1.75-1.764s.784-1.764 1.75-1.764 1.75.79 1.75 1.764-.783 1.764-1.75 1.764zm13.5 12.268h-3v-5.604c0-3.368-4-3.113-4 0v5.604h-3v-11h3v1.765c1.396-2.586 7-2.777 7 2.476v6.759z" clipRule="evenodd" />
            </svg>
          </Link>
        </div>
        <div className="mt-8 md:order-1 md:mt-0">
          <Link href="/" className="flex items-center">
            <Image
              src="/logo.svg"
            />
          </Link>
        </div>
      </div>
    </footer>
  );
}
```

```

        alt="Anic Digital"
        width={120}
        height={32}
        className="invert"
      />
    </Link>
    <p className="mt-4 text-xs leading-5 text-gray-400">
      &copy; {new Date().getFullYear()} Anic Digital. All rights reserved.
    </p>
  </div>
</div>
<div className="mx-auto max-w-7xl px-6 pb-8 lg:px-8">
  <div className="border-t border-gray-700 pt-8 grid grid-cols-2 gap-8 md:grid-
cols-4">
    <div>
      <h3 className="text-sm font-semibold leading-6 text-white">Company</
h3>
      <ul className="mt-6 space-y-4">
        <li>
          <Link href="/about" className="text-sm leading-6 text-gray-300
hover:text-white">
            About
          </Link>
        </li>
        <li>
          <Link href="/blog" className="text-sm leading-6 text-gray-300
hover:text-white">
            Blog
          </Link>
        </li>
        <li>
          <Link href="/contact" className="text-sm leading-6 text-gray-300
hover:text-white">
            Contact
          </Link>
        </li>
      </ul>
    </div>
    <div>
      <h3 className="text-sm font-semibold leading-6 text-white">Services</h3>
      <ul className="mt-6 space-y-4">
        <li>
          <Link href="/marketplace" className="text-sm leading-6 text-gray-300
hover:text-white">
            Marketplace
          </Link>
        </li>
        <li>
          <Link href="/pricing" className="text-sm leading-6 text-gray-300
hover:text-white">
            Pricing
          </Link>

```

```
</li>
</ul>
</div>
<div>
  <h3 className="text-sm font-semibold leading-6 text-white">Support</h3>
  <ul className="mt-6 space-y-4">
    <li>
      <Link href="/faq" className="text-sm leading-6 text-gray-300 hover:text-
white">
        FAQ
      </Link>
    </li>
    <li>
      <Link href="/terms" className="text-sm leading-6 text-gray-300
hover:text-white">
        Terms
      </Link>
    </li>
    <li>
      <Link href="/privacy" className="text-sm leading-6 text-gray-300
hover:text-white">
        Privacy
      </Link>
    </li>
  </ul>
</div>
<div>
  <h3 className="text-sm font-semibold leading-6 text-white">Account</h3>
  <ul className="mt-6 space-y-4">
    <li>
      <Link href="/login" className="text-sm leading-6 text-gray-300
hover:text-white">
        Login
      </Link>
    </li>
    <li>
      <Link href="/register" className="text-sm leading-6 text-gray-300
hover:text-white">
        Register
      </Link>
    </li>
    <li>
      <Link href="/dashboard" className="text-sm leading-6 text-gray-300
hover:text-white">
        Dashboard
      </Link>
    </li>
  </ul>
</div>
</div>
</div>
</div>
```

```
);  
}
```

## Dashboard Layout (src/app/(dashboard)/layout.tsx)

```
'use client';  
  
import { useSession } from 'next-auth/react';  
import { useRouter } from 'next/navigation';  
import { useEffect } from 'react';  
import Sidebar from '@components/layout/Sidebar';  
import Header from '@components/layout/Header';  
  
export default function DashboardLayout({  
  children,  
}: {  
  children: React.ReactNode;  
}) {  
  const { data: session, status } = useSession();  
  const router = useRouter();  
  
  useEffect(() => {  
    if (status === 'unauthenticated') {  
      router.push('/login');  
    }  
  }, [status, router]);  
  
  if (status === 'loading') {  
    return <div className="flex items-center justify-center min-h-screen">Loading...</div>;  
  }  
  
  return (  
    <div className="min-h-screen bg-gray-50">  
      <Header />  
      <div className="flex">  
        <Sidebar />  
        <main className="flex-1 p-6">{children}</main>  
      </div>  
    </div>  
  );  
}
```

## Sidebar Component (src/components/layout/Sidebar.tsx)

```
'use client';  
  
import Link from 'next/link';  
import { usePathname } from 'next/navigation';
```

```

import { useSession } from 'next-auth/react';
import {
  Home,
  ShoppingCart,
  Wallet,
  Settings,
  FileText,
  Tool,
  BarChart,
} from 'lucide-react';

export default function Sidebar() {
  const pathname = usePathname();
  const { data: session } = useSession();
  const isPublisher = session?.user?.role === 'PUBLISHER';
  const isAdmin = session?.user?.role === 'ADMIN';

  const customerNavigation = [
    { name: 'Dashboard', href: '/dashboard', icon: Home },
    { name: 'Orders', href: '/dashboard/orders', icon: ShoppingCart },
    { name: 'Wallet', href: '/dashboard/wallet', icon: Wallet },
    { name: 'Tools', href: '/dashboard/tools', icon: Tool },
    { name: 'Settings', href: '/dashboard/settings', icon: Settings },
  ];

  const publisherNavigation = [
    { name: 'Dashboard', href: '/dashboard', icon: Home },
    { name: 'Listings', href: '/publisher/listings', icon: FileText },
    { name: 'Orders', href: '/publisher/orders', icon: ShoppingCart },
    { name: 'Wallet', href: '/dashboard/wallet', icon: Wallet },
    { name: 'Settings', href: '/dashboard/settings', icon: Settings },
  ];

  const adminNavigation = [
    { name: 'Dashboard', href: '/admin', icon: Home },
    { name: 'Users', href: '/admin/users', icon: Home },
    { name: 'Listings', href: '/admin/listings', icon: FileText },
    { name: 'Orders', href: '/admin/orders', icon: ShoppingCart },
    { name: 'Transactions', href: '/admin/transactions', icon: Wallet },
    { name: 'Analytics', href: '/admin/analytics', icon: BarChart },
    { name: 'Settings', href: '/admin/settings', icon: Settings },
  ];

  const navigation = isAdmin
    ? adminNavigation
    : isPublisher
    ? publisherNavigation
    : customerNavigation;

  return (
    <div className="w-64 bg-white shadow-sm h-screen sticky top-0">
      <div className="h-full px-3 py-4 overflow-y-auto">

```

```

<ul className="space-y-2 font-medium">
  {navigation.map((item) => {
    const isActive = pathname === item.href;
    return (
      <li key={item.name}>
        <Link
          href={item.href}
          className={`flex items-center p-2 rounded-lg ${
            isActive
              ? 'bg-indigo-100 text-indigo-600'
              : 'text-gray-900 hover:bg-gray-100'
            }`}
        >
          <item.icon className={`w-5 h-5 ${isActive ? 'text-indigo-600' : 'text-gray-500'}`} />
          <span className="ml-3">{item.name}</span>
        </Link>
      </li>
    );
  })}
</ul>
</div>
</div>
);
}

```

## Authentication Components

### Login Page (src/app/(auth)/login/page.tsx)

```

'use client';

import { useState } from 'react';
import { useRouter } from 'next/navigation';
import Link from 'next/link';
import { signIn } from 'next-auth/react';
import { useForm } from 'react-hook-form';
import { zodResolver } from '@hookform/resolvers/zod';
import { z } from 'zod';
import Image from 'next/image';

const loginSchema = z.object({
  email: z.string().email('Please enter a valid email address'),
  password: z.string().min(6, 'Password must be at least 6 characters'),
});

type LoginFormValues = z.infer<typeof loginSchema>;

export default function LoginPage() {
  const router = useRouter();

```

```

const [error, setError] = useState<string | null>(null);
const [isLoading, setIsLoading] = useState(false);

const {
  register,
  handleSubmit,
  formState: { errors },
} = useForm<LoginFormValues>({
  resolver: zodResolver(loginSchema),
});

const onSubmit = async (data: LoginFormValues) => {
  setIsLoading(true);
  setError(null);

  try {
    const result = await signIn('credentials', {
      redirect: false,
      email: data.email,
      password: data.password,
    });

    if (result?.error) {
      setError('Invalid email or password');
      setIsLoading(false);
      return;
    }

    router.push('/dashboard');
  } catch (error) {
    setError('An unexpected error occurred. Please try again.');
```

```

    setIsLoading(false);
  }
};

return (
  <div className="flex min-h-screen flex-1">
    <div className="flex flex-1 flex-col justify-center px-4 py-12 sm:px-6 lg:flex-
none lg:px-20 xl:px-24">
      <div className="mx-auto w-full max-w-sm lg:w-96">
        <div>
          <Link href="/">
            <Image
              src="/logo.svg"
              alt="Anic Digital"
              width={150}
              height={40}
              priority
            />
          </Link>
          <h2 className="mt-8 text-2xl font-bold leading-9 tracking-tight text-
gray-900">
```

```

    Sign in to your account
  </h2>
  <p className="mt-2 text-sm leading-6 text-gray-500">
    Don't have an account?{' '}
    <Link
      href="/register"
      className="font-semibold text-indigo-600 hover:text-indigo-500"
    >
      Sign up
    </Link>
  </p>
</div>

<div className="mt-10">
  <div>
    <form onSubmit={handleSubmit(onSubmit)} className="space-y-6">
      {error && (
        <div className="rounded-md bg-red-50 p-4">
          <div className="flex">
            <div className="text-sm text-red-700">{error}</div>
          </div>
        </div>
      )}
    <div>
      <label
        htmlFor="email"
        className="block text-sm font-medium leading-6 text-gray-900"
      >
        Email address
      </label>
      <div className="mt-2">
        <input
          id="email"
          type="email"
          autoComplete="email"
          {...register('email')}
          className="block w-full rounded-md border-0 py-1.5 shadow-sm
ring-1 ring-inset ring-gray-300 placeholder:text-gray-400 focus:ring-2 focus:ring-
inset focus:ring-indigo-600 sm:text-sm sm:leading-6"
        />
        {errors.email && (
          <p className="mt-2 text-sm text-red-600">
            {errors.email.message}
          </p>
        )}
      </div>
    </div>

    <div>
      <label
        htmlFor="password"
        className="block text-sm font-medium leading-6 text-gray-900"

```



```

    >
    Password
  </label>
  <div className="mt-2">
    <input
      id="password"
      type="password"
      autoComplete="current-password"
      {...register('password')}
      className="block w-full rounded-md border-0 py-1.5 shadow-sm
ring-1 ring-inset ring-gray-300 placeholder:text-gray-400 focus:ring-2 focus:ring-
inset focus:ring-indigo-600 sm:text-sm sm:leading-6"
    />
    {errors.password && (
      <p className="mt-2 text-sm text-red-600">
        {errors.password.message}
      </p>
    )}
  </div>
</div>

<div className="flex items-center justify-between">
  <div className="flex items-center">
    <input
      id="remember-me"
      name="remember-me"
      type="checkbox"
      className="h-4 w-4 rounded border-gray-300 text-indigo-600
focus:ring-indigo-600"
    />
    <label
      htmlFor="remember-me"
      className="ml-3 block text-sm leading-6 text-gray-700"
    >
      Remember me
    </label>
  </div>

  <div className="text-sm leading-6">
    <Link
      href="/forgot-password"
      className="font-semibold text-indigo-600 hover:text-indigo-500"
    >
      Forgot password?
    </Link>
  </div>
</div>

<div>
  <button
    type="submit"
    disabled={isLoading}

```

```

        className="flex w-full justify-center rounded-md bg-indigo-600 px-3
py-1.5 text-sm font-semibold leading-6 text-white shadow-sm hover:bg-indigo-500
focus-visible:outline focus-visible:outline-2 focus-visible:outline-offset-2 focus-
visible:outline-indigo-600 disabled:opacity-50"
      >
        {isLoading ? 'Signing in...' : 'Sign in'}
      </button>
    </div>
  </form>
</div>
</div>
</div>
</div>
</div>
<div className="relative hidden w-0 flex-1 lg:block">
  <div className="absolute inset-0 h-full w-full bg-gradient-to-r from-
indigo-600 to-purple-600" />
</div>
</div>
);
}

```

## API Integration

### API Client (src/lib/api.ts)

```

import axios from 'axios';

const API_URL = process.env.NEXT_PUBLIC_API_URL;

const api = axios.create({
  baseURL: API_URL,
  headers: {
    'Content-Type': 'application/json',
  },
  withCredentials: true,
});

// Request interceptor for adding token
api.interceptors.request.use(
  (config) => {
    // You can add token from localStorage or cookies here if needed
    return config;
  },
  (error) => {
    return Promise.reject(error);
  }
);

// Response interceptor for handling errors
api.interceptors.response.use(

```

```

(response) => {
  return response;
},
async (error) => {
  const originalRequest = error.config;

  // Handle 401 errors (unauthorized)
  if (error.response?.status === 401 && !originalRequest._retry) {
    originalRequest._retry = true;

    try {
      // Refresh token logic could go here
      // For now, redirect to login
      window.location.href = '/login';
      return Promise.reject(error);
    } catch (refreshError) {
      return Promise.reject(refreshError);
    }
  }

  return Promise.reject(error);
}
);

export default api;

```

## Authentication API (src/lib/auth.ts)

```

import api from './api';

export interface LoginCredentials {
  email: string;
  password: string;
}

export interface RegisterData {
  email: string;
  password: string;
  firstName: string;
  lastName: string;
  role?: string;
}

export const authApi = {
  login: async (credentials: LoginCredentials) => {
    const response = await api.post('/auth/login', credentials);
    return response.data;
  },

  register: async (data: RegisterData) => {

```

```

    const response = await api.post('/auth/register', data);
    return response.data;
  },

  logout: async () => {
    const response = await api.post('/auth/logout');
    return response.data;
  },

  forgotPassword: async (email: string) => {
    const response = await api.post('/auth/forgot-password', { email });
    return response.data;
  },

  resetPassword: async (token: string, password: string) => {
    const response = await api.post('/auth/reset-password', {
      token,
      password,
    });
    return response.data;
  },

  getCurrentUser: async () => {
    const response = await api.get('/auth/me');
    return response.data;
  },
};

```

## NextAuth Configuration (src/app/api/auth/[...nextauth]/route.ts)

```

import NextAuth from 'next-auth';
import CredentialsProvider from 'next-auth/providers/credentials';
import { authApi } from '@/lib/auth';

const handler = NextAuth({
  providers: [
    CredentialsProvider({
      name: 'Credentials',
      credentials: {
        email: { label: 'Email', type: 'email' },
        password: { label: 'Password', type: 'password' },
      },
      async authorize(credentials) {
        if (!credentials?.email || !credentials?.password) {
          return null;
        }

        try {
          const response = await authApi.login({
            email: credentials.email,

```

```

    password: credentials.password,
  });

  if (response.success && response.data.user) {
    return {
      id: response.data.user.id,
      email: response.data.user.email,
      name: `${response.data.user.firstName} ${response.data.user.lastName}`,
      role: response.data.user.role,
      token: response.data.token,
    };
  }
  return null;
} catch (error) {
  console.error('Auth error:', error);
  return null;
}
},
}),
],
callbacks: {
  async jwt({ token, user }) {
    if (user) {
      token.id = user.id;
      token.email = user.email;
      token.name = user.name;
      token.role = user.role;
      token.accessToken = user.token;
    }
    return token;
  },
  async session({ session, token }) {
    if (token) {
      session.user = {
        id: token.id as string,
        email: token.email as string,
        name: token.name as string,
        role: token.role as string,
      };
      session.accessToken = token.accessToken as string;
    }
    return session;
  },
},
pages: {
  signIn: '/login',
  error: '/login',
},
session: {
  strategy: 'jwt',
  maxAge: 24 * 60 * 60, // 1 day
},

```

```

    secret: process.env.NEXTAUTH_SECRET,
  });

  export { handler as GET, handler as POST };

```

## Home Page (src/app/page.tsx)

```

import Link from 'next/link';
import Image from 'next/image';
import Header from '@components/layout/Header';
import Footer from '@components/layout/Footer';

export default function HomePage() {
  return (
    <div className="min-h-screen flex flex-col">
      <Header />
      <main className="flex-grow">
        { /* Hero Section */ }
        <div className="relative isolate overflow-hidden bg-gradient-to-b from-indigo-100/20">
          <div className="mx-auto max-w-7xl pb-24 pt-10 sm:pb-32 lg:grid lg:grid-cols-2 lg:gap-x-8 lg:px-8 lg:py-40">
            <div className="px-6 lg:px-0 lg:pt-4">
              <div className="mx-auto max-w-2xl">
                <div className="max-w-lg">
                  <h1 className="mt-10 text-4xl font-bold tracking-tight text-gray-900 sm:text-6xl">
                    Build Quality Backlinks with Ease
                  </h1>
                  <p className="mt-6 text-lg leading-8 text-gray-600">
                    Anic Digital connects you with high-quality websites for effective link building. Boost your SEO with our curated marketplace of publishers.
                  </p>
                  <div className="mt-10 flex items-center gap-x-6">
                    <Link
                      href="/marketplace"
                      className="rounded-md bg-indigo-600 px-3.5 py-2.5 text-sm font-semibold text-white shadow-sm hover:bg-indigo-500 focus-visible:outline focus-visible:outline-2 focus-visible:outline-offset-2 focus-visible:outline-indigo-600"
                    >
                      Browse Marketplace
                    </Link>
                    <Link
                      href="/about"
                      className="text-sm font-semibold leading-6 text-gray-900"
                    >
                      Learn more <span aria-hidden="true">→</span>
                    </Link>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </main>
    </div>
  );
}

```

```

    </div>
  </div>
  <div className="mt-20 sm:mt-24 md:mx-auto md:max-w-2xl lg:mx-0
lg:mt-0 lg:w-screen">
    <div className="shadow-xl md:rounded-3xl">
      <div className="bg-indigo-500 [clip-path:inset(0)] md:[clip-
path:inset(0_round_theme(borderRadius.3xl))]">
        <div className="absolute -inset-y-px left-1/2 -z-10 ml-10 w-[200%] skew-
x-[-30deg] bg-indigo-100 opacity-20 ring-1 ring-inset ring-white md:ml-20
lg:ml-36" />
        <div className="relative px-6 pt-8 sm:pt-16 md:pl-16 md:pr-0">
          <div className="mx-auto max-w-2xl md:mx-0 md:max-w-none">
            <div className="w-screen overflow-hidden rounded-tl-xl bg-
gray-900">
              <div className="flex bg-gray-800/40 ring-1 ring-white/5">
                <div className="-mb-px flex text-sm font-medium leading-6 text-
gray-400">
                  <div className="border-b border-r border-b-white/20 border-r-
white/10 bg-white/5 px-4 py-2 text-white">
                    Marketplace.jsx
                  </div>
                  <div className="border-r border-gray-600/10 px-4 py-2">
                    Dashboard.jsx
                  </div>
                </div>
              </div>
            <div className="px-6 pb-14 pt-6">
              /* Code snippet or screenshot could go here */
              <div className="text-white">
                <pre className="text-xs">
                  <code>
                    `export default function Marketplace() {
const [listings, setListings] = useState([]);

// Fetch listings from API
useEffect(() => {
  async function fetchListings() {
    const data = await api.get('/listings');
    setListings(data.listings);
  }
  fetchListings();
}, []);

return (
  <div className="grid grid-cols-3 gap-4">
    {listings.map((listing) => (
      <ListingCard key={listing.id} listing={listing} />
    ))}
  </div>
);
}}
                  </code>

```

```

    </pre>
  </div>
</div>
</div>
</div>
</div>
<div className="pointer-events-none absolute inset-0 ring-1 ring-inset
ring-black/10 md:rounded-3xl" />
  </div>
</div>
</div>
</div>
</div>
</div>

{/* Features Section */}
<div className="bg-white py-24 sm:py-32">
  <div className="mx-auto max-w-7xl px-6 lg:px-8">
    <div className="mx-auto max-w-2xl lg:text-center">
      <h2 className="text-base font-semibold leading-7 text-indigo-600">
        Boost Your SEO
      </h2>
      <p className="mt-2 text-3xl font-bold tracking-tight text-gray-900
sm:text-4xl">
        Everything you need for effective link building
      </p>
      <p className="mt-6 text-lg leading-8 text-gray-600">
        Our platform connects you with high-quality websites across various
niches, helping you build a strong backlink profile and improve your search engine
rankings.
      </p>
    </div>
    <div className="mx-auto mt-16 max-w-2xl sm:mt-20 lg:mt-24 lg:max-w-
none">
      <dl className="grid max-w-xl grid-cols-1 gap-x-8 gap-y-16 lg:max-w-none
lg:grid-cols-3">
        <div className="flex flex-col">
          <dt className="flex items-center gap-x-3 text-base font-semibold
leading-7 text-gray-900">
            <div className="h-5 w-5 flex-none text-indigo-600">
              <svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 20 20"
fill="currentColor" aria-hidden="true">
                <path fillRule="evenodd" d="M16.704 4.153a.75.75 0 01.143 1.052l-.8
10.5a.75.75 0 01-1.127.075l-4.5-4.5a.75.75 0 011.06-1.06l3.894 3.893 7.48-9.817a.
75.75 0 011.05-.143z" clipRule="evenodd" />
              </svg>
            </div>
            Curated Marketplace
          </dt>
          <dd className="mt-4 flex flex-auto flex-col text-base leading-7 text-
gray-600">
            <p className="flex-auto">
              Browse our curated marketplace of high-quality websites with verified

```



metrics like DA, DR, and traffic.

```
</p>
</dd>
</div>
<div className="flex flex-col">
  <dt className="flex items-center gap-x-3 text-base font-semibold
leading-7 text-gray-900">
    <div className="h-5 w-5 flex-none text-indigo-600">
      <svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 20 20"
fill="currentColor" aria-hidden="true">
        <path fillRule="evenodd" d="M16.704 4.153a.75.75 0 01.143 1.052l-.8
10.5a.75.75 0 01-1.127.075l-4.5-4.5a.75.75 0 011.06-1.06l3.894 3.893 7.48-9.817a.
75.75 0 011.05-.143z" clipRule="evenodd" />

```

```
</svg>
```

```
</div>
```

Streamlined Process

```
</dt>
<dd className="mt-4 flex flex-auto flex-col text-base leading-7 text-
gray-600">
```

```
<p className="flex-auto">
  Simple order process with content submission, revisions, and
publication tracking all in one place.
```

```
</p>
</dd>
</div>
<div className="flex flex-col">
  <dt className="flex items-center gap-x-3 text-base font-semibold
leading-7 text-gray-900">
    <div className="h-5 w-5 flex-none text-indigo-600">
      <svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 20 20"
fill="currentColor" aria-hidden="true">
        <path fillRule="evenodd" d="M16.704 4.153a.75.75 0 01.143 1.052l-.8
10.5a.75.75 0 01-1.127.075l-4.5-4.5a.75.75 0 011.06-1.06l3.894 3.893 7.48-9.817a.
75.75 0 011.05-.143z" clipRule="evenodd" />

```

```
</svg>
```

```
</div>
```

Advanced Tools

```
</dt>
<dd className="mt-4 flex flex-auto flex-col text-base leading-7 text-
gray-600">
```

```
<p className="flex-auto">
  Access competitor analysis tools, SEO metrics tracking, and
performance reporting.
```

```
</p>
</dd>
</div>
</dl>
</div>
</div>
</div>
```

```
{/* CTA Section */}
```

```

    <div className="bg-indigo-600">
      <div className="mx-auto max-w-7xl px-6 py-24 sm:py-32 lg:flex lg:items-
center lg:justify-between lg:px-8">
        <h2 className="text-3xl font-bold tracking-tight text-white sm:text-4xl">
          Ready to boost your SEO?
        <br />
        Start building quality backlinks today.
      </h2>
      <div className="mt-10 flex items-center gap-x-6 lg:mt-0 lg:flex-shrink-0">
        <Link
          href="/register"
          className="rounded-md bg-white px-3.5 py-2.5 text-sm font-semibold
text-indigo-600 shadow-sm hover:bg-indigo-50 focus-visible:outline focus-
visible:outline-2 focus-visible:outline-offset-2 focus-visible:outline-white"
        >
          Get started
        </Link>
        <Link
          href="/contact"
          className="text-sm font-semibold leading-6 text-white"
        >
          Contact sales <span aria-hidden="true">→</span>
        </Link>
      </div>
    </div>
  </div>
</main>
<Footer />
</div>
);
}

```

## Dashboard Page (src/app/(dashboard)/page.tsx)

```

'use client';

import { useEffect, useState } from 'react';
import { useSession } from 'next-auth/react';
import Link from 'next/link';
import api from '@lib/api';

interface DashboardStats {
  activeOrders: number;
  completedOrders: number;
  walletBalance: number;
  recentActivity: Array<{
    id: string;
    type: string;
    description: string;
    date: string;
  }>
}

```

```
}>;  
}
```

```
export default function DashboardPage() {  
  const { data: session } = useSession();  
  const [stats, setStats] = useState<DashboardStats | null>(null);  
  const [isLoading, setIsLoading] = useState(true);
```

```
  useEffect(() => {  
    const fetchDashboardStats = async () => {  
      try {  
        // This would be replaced with actual API call  
        // const response = await api.get('/dashboard/stats');  
        // setStats(response.data);  
  
        // Mock data for now  
        setStats({  
          activeOrders: 3,  
          completedOrders: 12,  
          walletBalance: 250.00,  
          recentActivity: [  
            {  
              id: '1',  
              type: 'order',  
              description: 'Order #1234 completed',  
              date: '2023-05-15T10:30:00Z',  
            },  
            {  
              id: '2',  
              type: 'payment',  
              description: 'Wallet topped up with $100',  
              date: '2023-05-14T14:20:00Z',  
            },  
            {  
              id: '3',  
              type: 'order',  
              description: 'New order #1235 created',  
              date: '2023-05-13T09:15:00Z',  
            },  
          ],  
        });  
  
        setIsLoading(false);  
      } catch (error) {  
        console.error('Error fetching dashboard stats:', error);  
        setIsLoading(false);  
      }  
    };  
  
    fetchDashboardStats();  
  }, []);
```



```

<div className="flex items-center">
  <div className="flex-shrink-0 bg-green-500 rounded-md p-3">
    <svg className="h-6 w-6 text-white" fill="none" viewBox="0 0 24 24"
stroke="currentColor">
      <path strokeLinecap="round" strokeLinejoin="round" strokeWidth={2}
d="M12 8c-1.657 0-3 .895-3 2s1.343 2 3 2 3 .895 3 2-1.343 2-3 2m0-8c1.11 0
2.08.402 2.599 1M12 8V7m0 1v8m0 0v1m0-1c-1.11 0-2.08-.402-2.599-1M21 12a9 9
0 11-18 0 9 9 0 0118 0z" />
    </svg>
  </div>
  <div className="ml-5 w-0 flex-1">
    <dl>
      <dt className="text-sm font-medium text-gray-500 truncate">
        Wallet Balance
      </dt>
      <dd className="flex items-baseline">
        <div className="text-2xl font-semibold text-gray-900">
          ${stats?.walletBalance.toFixed(2)}
        </div>
      </dd>
    </dl>
  </div>
</div>
<div className="bg-gray-50 px-4 py-4 sm:px-6">
  <div className="text-sm">
    <Link href="/dashboard/wallet" className="font-medium text-indigo-600
hover:text-indigo-500">
      Manage wallet
    </Link>
  </div>
</div>
</div>

<div className="bg-white overflow-hidden shadow rounded-lg">
  <div className="px-4 py-5 sm:p-6">
    <div className="flex items-center">
      <div className="flex-shrink-0 bg-purple-500 rounded-md p-3">
        <svg className="h-6 w-6 text-white" fill="none" viewBox="0 0 24 24"
stroke="currentColor">
          <path strokeLinecap="round" strokeLinejoin="round" strokeWidth={2}
d="M9 5H7a2 2 0 0-2 2v12a2 2 0 02 2h10a2 2 0 02-2V7a2 2 0 0-2 2h-2M9 5a2 2
0 02 2h2a2 2 0 02-2M9 5a2 2 0 012-2h2a2 2 0 012 2" />
        </svg>
      </div>
      <div className="ml-5 w-0 flex-1">
        <dl>
          <dt className="text-sm font-medium text-gray-500 truncate">
            Completed Orders
          </dt>
          <dd className="flex items-baseline">
            <div className="text-2xl font-semibold text-gray-900">

```



```
<div className="px-4 py-5 sm:p-6">
  <h3 className="text-lg leading-6 font-medium text-gray-900">
    Quick Actions
  </h3>
  <div className="mt-5 grid grid-cols-1 gap-4 sm:grid-cols-2">
    <div>
      <Link
        href="/marketplace"
        className="w-full inline-flex items-center justify-center px-4 py-2 border
border-transparent text-sm font-medium rounded-md shadow-sm text-white bg-
indigo-600 hover:bg-indigo-700 focus:outline-none focus:ring-2 focus:ring-offset-2
focus:ring-indigo-500"
      >
        Browse Marketplace
      </Link>
    </div>
    <div>
      <Link
        href="/dashboard/wallet/topup"
        className="w-full inline-flex items-center justify-center px-4 py-2 border
border-gray-300 shadow-sm text-sm font-medium rounded-md text-gray-700 bg-
white hover:bg-gray-50 focus:outline-none focus:ring-2 focus:ring-offset-2
focus:ring-indigo-500"
      >
        Top Up Wallet
      </Link>
    </div>
    <div>
      <Link
        href="/dashboard/tools/competitor-analysis"
        className="w-full inline-flex items-center justify-center px-4 py-2 border
border-gray-300 shadow-sm text-sm font-medium rounded-md text-gray-700 bg-
white hover:bg-gray-50 focus:outline-none focus:ring-2 focus:ring-offset-2
focus:ring-indigo-500"
      >
        Competitor Analysis
      </Link>
    </div>
    <div>
      <Link
        href="/dashboard/settings"
        className="w-full inline-flex items-center justify-center px-4 py-2 border
border-gray-300 shadow-sm text-sm font-medium rounded-md text-gray-700 bg-
white hover:bg-gray-50 focus:outline-none focus:ring-2 focus:ring-offset-2
focus:ring-indigo-500"
      >
        Account Settings
      </Link>
    </div>
  </div>
</div>
```

```

    <div className="bg-white shadow sm:rounded-lg">
      <div className="px-4 py-5 sm:p-6">
        <h3 className="text-lg leading-6 font-medium text-gray-900">
          Need Help?
        </h3>
        <div className="mt-2 max-w-xl text-sm text-gray-500">
          <p>
            Our support team is here to help you with any questions or issues you
            may have.
          </p>
        </div>
        <div className="mt-5">
          <Link
            href="/contact"
            className="inline-flex items-center justify-center px-4 py-2 border
            border-transparent text-sm font-medium rounded-md shadow-sm text-white bg-
            indigo-600 hover:bg-indigo-700 focus:outline-none focus:ring-2 focus:ring-offset-2
            focus:ring-indigo-500"
          >
            Contact Support
          </Link>
        </div>
      </div>
    </div>
  </div>
);
}

```

## Marketplace Page (src/app/(marketplace)/page.tsx)

```

'use client';

import { useEffect, useState } from 'react';
import Link from 'next/link';
import { useSearchParams } from 'next/navigation';
import api from '@lib/api';
import FilterPanel from '@components/marketplace/FilterPanel';
import ListingCard from '@components/marketplace/ListingCard';
import SearchBar from '@components/marketplace/SearchBar';
import Header from '@components/layout/Header';
import Footer from '@components/layout/Footer';

interface Listing {
  id: string;
  title: string;
  description: string;
  price: number;
  publisher: {

```



```

    id: string;
    firstName: string;
    lastName: string;
  };
  category: {
    id: string;
    name: string;
  };
  metrics: {
    da: number;
    dr: number;
    traffic: number;
    keywords: number;
    refDomains: number;
    country: string;
  };
}

```

```

interface Category {
  id: string;
  name: string;
}

```

```

export default function MarketplacePage() {
  const searchParams = useSearchParams();
  const [listings, setListings] = useState<Listing[]>([]);
  const [categories, setCategories] = useState<Category[]>([]);
  const [isLoading, setIsLoading] = useState(true);
  const [totalListings, setTotalListings] = useState(0);
  const [currentPage, setCurrentPage] = useState(1);
  const [totalPages, setTotalPages] = useState(1);

```

*// Get filter values from URL params*

```

const category = searchParams.get('category') || '';
const minPrice = searchParams.get('minPrice') || '';
const maxPrice = searchParams.get('maxPrice') || '';
const minDa = searchParams.get('minDa') || '';
const minDr = searchParams.get('minDr') || '';
const country = searchParams.get('country') || '';
const search = searchParams.get('search') || '';
const sort = searchParams.get('sort') || 'date:desc';
const page = searchParams.get('page') || '1';

```

```

useEffect(() => {
  const fetchListings = async () => {
    try {
      // This would be replaced with actual API call
      // const response = await api.get('/listings', {
      //   params: {
      //     category,
      //     minPrice,
      //     maxPrice,

```

```
// minDa,
// minDr,
// country,
// search,
// sort,
// page,
// limit: 9,
// },
// });
// setListings(response.data.listings);
// setTotalListings(response.data.pagination.total);
// setTotalPages(response.data.pagination.pages);

// Mock data for now
setListings([
  {
    id: '1',
    title: 'Technology Blog Guest Post',
    description: 'High-quality technology blog accepting guest posts on AI,
software development, and tech trends.',
    price: 150,
    publisher: {
      id: '101',
      firstName: 'John',
      lastName: 'Doe',
    },
    category: {
      id: '201',
      name: 'Technology',
    },
    metrics: {
      da: 45,
      dr: 52,
      traffic: 25000,
      keywords: 3500,
      refDomains: 230,
      country: 'US',
    },
  },
  {
    id: '2',
    title: 'Health and Wellness Blog',
    description: 'Popular health blog accepting guest posts on nutrition, fitness,
and mental health topics.',
    price: 120,
    publisher: {
      id: '102',
      firstName: 'Jane',
      lastName: 'Smith',
    },
    category: {
      id: '202',
```

```
    name: 'Health',
  },
  metrics: {
    da: 38,
    dr: 45,
    traffic: 18000,
    keywords: 2800,
    refDomains: 180,
    country: 'UK',
  },
},
{
  id: '3',
  title: 'Finance and Investment Blog',
  description: 'Established finance blog accepting guest posts on personal
finance, investing, and cryptocurrency.',
  price: 200,
  publisher: {
    id: '103',
    firstName: 'Robert',
    lastName: 'Johnson',
  },
  category: {
    id: '203',
    name: 'Finance',
  },
  metrics: {
    da: 52,
    dr: 58,
    traffic: 32000,
    keywords: 4200,
    refDomains: 310,
    country: 'US',
  },
},
{
  id: '4',
  title: 'Travel and Lifestyle Blog',
  description: 'Popular travel blog accepting guest posts on destinations,
travel tips, and lifestyle topics.',
  price: 130,
  publisher: {
    id: '104',
    firstName: 'Sarah',
    lastName: 'Williams',
  },
  category: {
    id: '204',
    name: 'Travel',
  },
  metrics: {
    da: 41,
```

```
dr: 47,  
traffic: 22000,  
keywords: 3100,  
refDomains: 210,  
country: 'AU',  
},  
},  
{  
  id: '5',  
  title: 'Marketing and Business Blog',  
  description: 'Business-focused blog accepting guest posts on marketing,  
entrepreneurship, and business strategy.',  
  price: 180,  
  publisher: {  
    id: '105',  
    firstName: 'Michael',  
    lastName: 'Brown',  
  },  
  category: {  
    id: '205',  
    name: 'Business',  
  },  
  metrics: {  
    da: 49,  
    dr: 54,  
    traffic: 28000,  
    keywords: 3800,  
    refDomains: 270,  
    country: 'US',  
  },  
},  
},  
{  
  id: '6',  
  title: 'Education and Learning Blog',  
  description: 'Educational blog accepting guest posts on learning methods,  
education technology, and career development.',  
  price: 110,  
  publisher: {  
    id: '106',  
    firstName: 'Emily',  
    lastName: 'Davis',  
  },  
  category: {  
    id: '206',  
    name: 'Education',  
  },  
  metrics: {  
    da: 36,  
    dr: 42,  
    traffic: 15000,  
    keywords: 2500,  
    refDomains: 160,
```

```
    country: 'CA',
  },
},
]);
```

```
setCategories([
  { id: '201', name: 'Technology' },
  { id: '202', name: 'Health' },
  { id: '203', name: 'Finance' },
  { id: '204', name: 'Travel' },
  { id: '205', name: 'Business' },
  { id: '206', name: 'Education' },
]);
```

```
setTotalListings(24);
setTotalPages(4);
setCurrentPage(parseInt(page));
setIsLoading(false);
} catch (error) {
  console.error('Error fetching listings:', error);
  setIsLoading(false);
}
};
```

```
fetchListings();
}, [category, minPrice, maxPrice, minDa, minDr, country, search, sort, page]);
```

```
return (
  <div className="min-h-screen flex flex-col">
    <Header />
    <main className="flex-grow bg-gray-50">
      <div className="max-w-7xl mx-auto px-4 sm:px-6 lg:px-8 py-8">
        <div className="mb-8">
          <h1 className="text-3xl font-bold text-gray-900">Marketplace</h1>
          <p className="mt-2 text-sm text-gray-500">
            Browse our curated selection of high-quality websites for your link
            building campaign.
          </p>
        </div>

        <div className="mb-6">
          <SearchBar initialValue={search} />
        </div>

        <div className="flex flex-col lg:flex-row gap-8">
          <div className="w-full lg:w-1/4">
            <FilterPanel
              categories={categories}
              initialValues={{
                category,
                minPrice,
                maxPrice,
```

```

        minDa,
        minDr,
        country,
        sort,
      }}
    />
  </div>

  <div className="w-full lg:w-3/4">
    {isLoading ? (
      <div className="flex justify-center items-center h-64">
        <div className="animate-spin rounded-full h-12 w-12 border-t-2 border-
b-2 border-indigo-500"></div>
      </div>
    ) : (
      <>
        <div className="mb-4 flex justify-between items-center">
          <p className="text-sm text-gray-500">
            Showing {listings.length} of {totalListings} listings
          </p>
          <div className="flex items-center">
            <label htmlFor="sort" className="text-sm text-gray-500 mr-2">
              Sort by:
            </label>
            <select
              id="sort"
              className="text-sm border-gray-300 rounded-md shadow-sm
focus:border-indigo-300 focus:ring focus:ring-indigo-200 focus:ring-opacity-50"
              value={sort}
              onChange={(e) => {
                const url = new URL(window.location.href);
                url.searchParams.set('sort', e.target.value);
                window.history.pushState({}, "", url);
                // This would trigger the useEffect to fetch listings
              }}
            >
              <option value="date:desc">Newest</option>
              <option value="price:asc">Price: Low to High</option>
              <option value="price:desc">Price: High to Low</option>
              <option value="da:desc">DA: High to Low</option>
              <option value="dr:desc">DR: High to Low</option>
              <option value="traffic:desc">Traffic: High to Low</option>
            </select>
          </div>
        </div>

        <div>

          <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-6">
            {listings.map((listing) => (
              <ListingCard key={listing.id} listing={listing} />
            ))}
          </div>

```

```

{totalPages > 1 && (
  <div className="mt-8 flex justify-center">
    <nav className="relative z-0 inline-flex rounded-md shadow-sm -
space-x-px" aria-label="Pagination">
      <Link
        href={` /marketplace?page=${Math.max(1, currentPage - 1)}`}
        className={`relative inline-flex items-center px-2 py-2 rounded-l-md
border border-gray-300 bg-white text-sm font-medium text-gray-500 hover:bg-
gray-50 ${
          currentPage === 1 ? 'cursor-not-allowed opacity-50' : ''
        }}`
      >
        <span className="sr-only">Previous</span>
        <svg className="h-5 w-5" xmlns="http://www.w3.org/2000/svg"
viewBox="0 0 20 20" fill="currentColor" aria-hidden="true">
          <path fillRule="evenodd" d="M12.707 5.293a1 1 0 010 1.414L9.414
10l3.293 3.293a1 1 0 01-1.414 1.414l-4-4a1 1 0 010-1.414l4-4a1 1 0 011.414 0z"
clipRule="evenodd" />
        </svg>
      </Link>

      {Array.from({ length: totalPages }, (_, i) => i + 1).map((page) => (
        <Link
          key={page}
          href={` /marketplace?page=${page}`}
          className={`relative inline-flex items-center px-4 py-2 border text-
sm font-medium ${
            currentPage === page
              ? 'z-10 bg-indigo-50 border-indigo-500 text-indigo-600'
              : 'bg-white border-gray-300 text-gray-500 hover:bg-gray-50'
          }}`
        >
          {page}
        </Link>
      ))}

      <Link
        href={` /marketplace?page=${Math.min(totalPages, currentPage + 1)}`}
      >
        <span className="sr-only">Next</span>
        <svg className="h-5 w-5" xmlns="http://www.w3.org/2000/svg"
viewBox="0 0 20 20" fill="currentColor" aria-hidden="true">
          <path fillRule="evenodd"
d="M7.293 14.707a1 1 0 010 1.414L10.586 10 7.293 6.707a1 1 0 011.414-1.414l4
4a1 1 0 010 1.414l-4 4a1 1 0 01-1.414 0z" clipRule="evenodd" />
        </svg>
      </div>
    <div className={`relative inline-flex items-center px-2 py-2 rounded-r-md border
border-gray-300 bg-white text-sm font-medium text-gray-500 hover:bg-gray-50 ${
      currentPage === totalPages ? 'cursor-not-allowed opacity-50' : ''
    }}`
    >
      <span className="sr-only">Next</span>
      <svg className="h-5 w-5" xmlns="http://www.w3.org/2000/svg"
viewBox="0 0 20 20" fill="currentColor" aria-hidden="true">
        <path fillRule="evenodd"
d="M7.293 14.707a1 1 0 010 1.414L10.586 10 7.293 6.707a1 1 0 011.414-1.414l4
4a1 1 0 010 1.414l-4 4a1 1 0 01-1.414 0z" clipRule="evenodd" />
        </svg>
      </div>
    </div>
  </div>
)

```

```

        </Link>
      </nav>
    </div>
  )}
</>
  )}
</div>
</div>
</div>
</main>
<Footer />
</div>
);
}

```

## Marketplace Components

### ListingCard Component (src/components/marketplace/ListingCard.tsx)

```

import Link from 'next/link';

interface Listing {
  id: string;
  title: string;
  description: string;
  price: number;
  publisher: {
    id: string;
    firstName: string;
    lastName: string;
  };
  category: {
    id: string;
    name: string;
  };
  metrics: {
    da: number;
    dr: number;
    traffic: number;
    keywords: number;
    refDomains: number;
    country: string;
  };
}

interface ListingCardProps {
  listing: Listing;
}

export default function ListingCard({ listing }: ListingCardProps) {

```





```

        <span className="ml-1 text-sm font-semibold text-gray-900">{listing.metrics.country}</span>
      </div>
    </div>
  </div>
</div>

<div className="bg-gray-50 px-6 py-4">
  <div className="flex items-center justify-between">
    <div className="flex items-center">
      <div className="flex-shrink-0">
        <span className="inline-block h-8 w-8 rounded-full bg-indigo-100 text-indigo-700 text-center leading-8">
          {listing.publisher.firstName.charAt(0)}
          {listing.publisher.lastName.charAt(0)}
        </span>
      </div>
      <div className="ml-3">
        <p className="text-sm font-medium text-gray-900">
          {listing.publisher.firstName} {listing.publisher.lastName}
        </p>
      </div>
    </div>
    <Link
      href={` /marketplace/${listing.id}`}
      className="inline-flex items-center px-3 py-1 border border-transparent text-sm leading-4 font-medium rounded-md shadow-sm text-white bg-indigo-600 hover:bg-indigo-700 focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-indigo-500"
    >
      View Details
    </Link>
  </div>
</div>
</div>
);
}

```

## FilterPanel Component (src/components/marketplace/FilterPanel.tsx)

```

'use client';

import { useState, useEffect } from 'react';
import { useRouter } from 'next/navigation';

interface Category {
  id: string;
  name: string;
}

```

```

interface FilterPanelProps {
  categories: Category[];
  initialValues: {
    category: string;
    minPrice: string;
    maxPrice: string;
    minDa: string;
    minDr: string;
    country: string;
    sort: string;
  };
}

```

```

export default function FilterPanel({ categories, initialValues }: FilterPanelProps) {
  const router = useRouter();
  const [filters, setFilters] = useState({
    category: initialValues.category,
    minPrice: initialValues.minPrice,
    maxPrice: initialValues.maxPrice,
    minDa: initialValues.minDa,
    minDr: initialValues.minDr,
    country: initialValues.country,
  });

```

```

  const handleChange = (e: React.ChangeEvent<HTMLInputElement | HTMLSelectElement>) => {
    const { name, value } = e.target;
    setFilters((prev) => ({ ...prev, [name]: value }));
  };

```

```

const applyFilters = () => {
  const searchParams = new URLSearchParams();

```

```

  Object.entries(filters).forEach(([key, value]) => {
    if (value) {
      searchParams.set(key, value);
    }
  });

```

*// Preserve sort and search parameters*

```

const currentUrl = new URL(window.location.href);
if (currentUrl.searchParams.has('sort')) {
  searchParams.set('sort', currentUrl.searchParams.get('sort')!);
}
if (currentUrl.searchParams.has('search')) {
  searchParams.set('search', currentUrl.searchParams.get('search')!);
}

```

*// Reset to page 1 when filters change*

```

searchParams.set('page', '1');

router.push(`/marketplace?${searchParams.toString()}`);

```

```

};

const resetFilters = () => {
  setFilters({
    category: "",
    minPrice: "",
    maxPrice: "",
    minDa: "",
    minDr: "",
    country: "",
  });

  // Preserve sort and search parameters
  const searchParams = new URLSearchParams();
  const currentUrl = new URL(window.location.href);
  if (currentUrl.searchParams.has('sort')) {
    searchParams.set('sort', currentUrl.searchParams.get('sort')!);
  }
  if (currentUrl.searchParams.has('search')) {
    searchParams.set('search', currentUrl.searchParams.get('search')!);
  }

  router.push(`/marketplace?${searchParams.toString()}`);
};

return (
  <div className="bg-white shadow rounded-lg p-6">
    <h2 className="text-lg font-medium text-gray-900 mb-4">Filters</h2>

    <div className="space-y-4">
      <div>
        <label htmlFor="category" className="block text-sm font-medium text-gray-700">
          Category
        </label>
        <select
          id="category"
          name="category"
          className="mt-1 block w-full pl-3 pr-10 py-2 text-base border-gray-300
focus:outline-none focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm
rounded-md"
          value={filters.category}
          onChange={handleChange}
        >
          <option value="">All Categories</option>
          {categories.map((category) => (
            <option key={category.id} value={category.id}>
              {category.name}
            </option>
          ))}
        </select>
      </div>

```

```
<div>
  <label htmlFor="minPrice" className="block text-sm font-medium text-
gray-700">
    Min Price ($)
  </label>
  <input
    type="number"
    name="minPrice"
    id="minPrice"
    className="mt-1 block w-full border-gray-300 rounded-md shadow-sm
focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm"
    placeholder="Min Price"
    value={filters.minPrice}
    onChange={handleChange}
  />
</div>
```

```
<div>
  <label htmlFor="maxPrice" className="block text-sm font-medium text-
gray-700">
    Max Price ($)
  </label>
  <input
    type="number"
    name="maxPrice"
    id="maxPrice"
    className="mt-1 block w-full border-gray-300 rounded-md shadow-sm
focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm"
    placeholder="Max Price"
    value={filters.maxPrice}
    onChange={handleChange}
  />
</div>
```

```
<div>
  <label htmlFor="minDa" className="block text-sm font-medium text-
gray-700">
    Min Domain Authority (DA)
  </label>
  <input
    type="number"
    name="minDa"
    id="minDa"
    className="mt-1 block w-full border-gray-300 rounded-md shadow-sm
focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm"
    placeholder="Min DA"
    value={filters.minDa}
    onChange={handleChange}
  />
</div>
```

```

<div>
  <label htmlFor="minDr" className="block text-sm font-medium text-
gray-700">
    Min Domain Rating (DR)
  </label>
  <input
    type="number"
    name="minDr"
    id="minDr"
    className="mt-1 block w-full border-gray-300 rounded-md shadow-sm
focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm"
    placeholder="Min DR"
    value={filters.minDr}
    onChange={handleChange}
  />
</div>

<div>
  <label htmlFor="country" className="block text-sm font-medium text-
gray-700">
    Country
  </label>
  <select
    id="country"
    name="country"
    className="mt-1 block w-full pl-3 pr-10 py-2 text-base border-gray-300
focus:outline-none focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm
rounded-md"
    value={filters.country}
    onChange={handleChange}
  >
    <option value="">All Countries</option>
    <option value="US">United States</option>
    <option value="UK">United Kingdom</option>
    <option value="CA">Canada</option>
    <option value="AU">Australia</option>
    <option value="DE">Germany</option>
    <option value="FR">France</option>
    <option value="IN">India</option>
  </select>
</div>

<div className="pt-4 flex flex-col space-y-2">
  <button
    type="button"
    onClick={applyFilters}
    className="w-full inline-flex justify-center items-center px-4 py-2 border
border-transparent text-sm font-medium rounded-md shadow-sm text-white bg-
indigo-600 hover:bg-indigo-700 focus:outline-none focus:ring-2 focus:ring-offset-2
focus:ring-indigo-500"
  >
    Apply Filters

```

```

    </button>
    <button
      type="button"
      onClick={resetFilters}
      className="w-full inline-flex justify-center items-center px-4 py-2 border
border-gray-300 shadow-sm text-sm font-medium rounded-md text-gray-700 bg-
white hover:bg-gray-50 focus:outline-none focus:ring-2 focus:ring-offset-2
focus:ring-indigo-500"
    >
      Reset Filters
    </button>
  </div>
</div>
</div>
);
}

```

## SearchBar Component (src/components/marketplace/SearchBar.tsx)

```

'use client';

import { useState } from 'react';
import { useRouter } from 'next/navigation';
import { Search } from 'lucide-react';

interface SearchBarProps {
  initialValue: string;
}

export default function SearchBar({ initialValue }: SearchBarProps) {
  const router = useRouter();
  const [searchTerm, setSearchTerm] = useState(initialValue);

  const handleSearch = (e: React.FormEvent) => {
    e.preventDefault();

    const searchParams = new URLSearchParams();
    if (searchTerm) {
      searchParams.set('search', searchTerm);
    }

    // Preserve other parameters
    const currentUrl = new URL(window.location.href);
    ['category', 'minPrice', 'maxPrice', 'minDa', 'minDr', 'country',
'sort'].forEach((param) => {
      if (currentUrl.searchParams.has(param)) {
        searchParams.set(param, currentUrl.searchParams.get(param)!);
      }
    });
  };

```

```

// Reset to page 1 when search changes
searchParams.set('page', '1');

router.push(`/marketplace?${searchParams.toString()}`);
};

return (
  <form onSubmit={handleSearch} className="relative">
    <div className="flex rounded-md shadow-sm">
      <div className="relative flex-grow focus-within:z-10">
        <div className="absolute inset-y-0 left-0 pl-3 flex items-center pointer-
events-none">
          <Search className="h-5 w-5 text-gray-400" aria-hidden="true" />
        </div>
        <input
          type="text"
          name="search"
          id="search"
          className="block w-full rounded-md border-gray-300 pl-10 focus:border-
indigo-500 focus:ring-indigo-500 sm:text-sm"
          placeholder="Search for websites, niches, or keywords"
          value={searchTerm}
          onChange={(e) => setSearchTerm(e.target.value)}
        />
      </div>
      <button
        type="submit"
        className="-ml-px relative inline-flex items-center space-x-2 px-4 py-2 border
border-gray-300 text-sm font-medium rounded-r-md text-gray-700 bg-gray-50
hover:bg-gray-100 focus:outline-none focus:ring-1 focus:ring-indigo-500
focus:border-indigo-500"
      >
        Search
      </button>
    </div>
  </form>
);
}

```

## Next Steps

After implementing this basic frontend setup, the next steps would be:

1. **Complete the remaining pages:**
2. Order management pages
3. Wallet and payment pages
4. User settings pages



5. Publisher dashboard pages
6. Admin panel pages
7. **Implement state management:**
8. Set up Zustand stores for client-side state
9. Implement React Query hooks for server state
10. **Connect to backend API:**
11. Replace mock data with actual API calls
12. Implement error handling and loading states
13. **Add authentication flow:**
14. Complete NextAuth integration
15. Implement protected routes
16. Add role-based access control
17. **Implement form validation:**
18. Add Zod schemas for all forms
19. Implement client-side validation
20. **Add responsive design:**
21. Ensure all pages work well on mobile devices
22. Optimize for different screen sizes
23. **Implement SEO metrics integration:**
24. Connect to third-party APIs
25. Display metrics data in the UI
26. **Add testing:**
27. Write unit tests for components
28. Add integration tests for key flows

This basic setup provides the foundation for building out the complete frontend application. The structure follows best practices for Next.js applications and includes all the necessary components for a functional marketplace platform.