



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO



Nombre: C. Hernandez Aguilar Ludwing Mauricio

Numero de Control: 211130222

Turno: Matutino

Semestre: 9no Semestre

Carrera: Ingeniería en Tecnologías de la Información
y Comunicación.

Profesor: Mtro. Nestor Eduardo Mondragon

Materia: Desarrolla Aplicaciones CSR

Actividad: Mini red social

Introducción

En el desarrollo web moderno, la forma en que el contenido se entrega a los usuarios es clave. La estrategia de renderizado puede impactar directamente la velocidad del sitio, la experiencia de usuario y el posicionamiento en motores de búsqueda. Por ello, es fundamental conocer las diferencias entre Client-Side Rendering (CSR), Server-Side Rendering (SSR) y Static Site Generation (SSG).

En este proyecto, nos enfocaremos en desarrollar una mini red social utilizando CSR. Para ello, emplearemos un stack de tecnologías completo: React.js para el frontend, Express para el backend y MongoDB como base de datos. Además, integraremos herramientas como Socket.IO para la comunicación en tiempo real y Tailwind CSS para un diseño rápido y eficiente.

Desarrollo

Client-Side Rendering (CSR)

El Client-Side Rendering se refiere a cuando el contenido de una página web se genera directamente en el navegador del usuario mediante JavaScript. Esto significa que el servidor entrega al navegador un archivo base, normalmente HTML vacío, junto con los scripts necesarios para construir la interfaz completa. Una vez que el navegador ejecuta estos scripts, se genera dinámicamente todo el contenido visible. La ventaja de CSR es que permite experiencias muy interactivas y dinámicas, ideales para aplicaciones tipo SPA (Single Page Application), pero puede generar tiempos de carga inicial más largos y afectar la indexación por motores de búsqueda si no se implementa correctamente.

Creación de backend

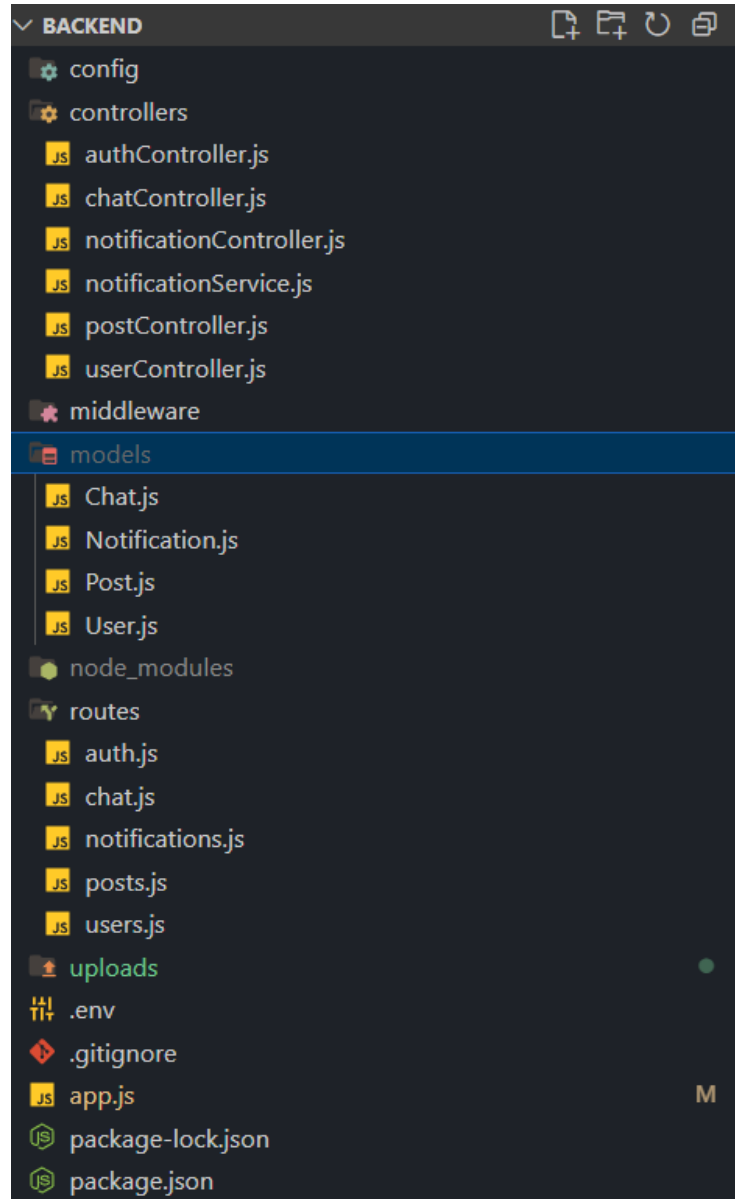
```
Directory: C:\Users\AizenLM\desktop
Mode                LastWriteTime         Length Name
----                -
d-----          9/23/2025   9:29 PM                red-social
PS C:\Users\AizenLM\desktop> cd .\red-social\
PS C:\Users\AizenLM\desktop\red-social> mkdir backend
Directory: C:\Users\AizenLM\desktop\red-social
Mode                LastWriteTime         Length Name
----                -
d-----          9/23/2025   9:30 PM                backend
PS C:\Users\AizenLM\desktop\red-social> cd backend
PS C:\Users\AizenLM\desktop\red-social\backend> npm init -y
Wrote to C:\Users\AizenLM\desktop\red-social\backend\package.json:
{
  "name": "backend",
```

```
"version": "1.0.0",
"description": "",
"main": "index.js",
"scripts": {
  "test": "echo \"Error: no test specified\" && exit 1"
},
"keywords": [],
"author": "",
"license": "ISC",
"type": "commonjs"
}
```

Instalación de dependencias

```
{
  "name": "mini-social",
  "version": "1.0.0",
  "description": "Mini red Social",
  "keywords": [
    "socket.io",
    "express",
    "mongoose"
  ],
  "license": "ISC",
  "author": "LudwingHA",
  "type": "module",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "server": "nodemon app.js "
  },
  "dependencies": {
    "bcryptjs": "^3.0.2",
    "cors": "^2.8.5",
    "express": "^5.1.0",
    "jsonwebtoken": "^9.0.2",
    "mongoose": "^8.18.1",
    "multer": "^2.0.2",
    "socket.io": "^4.8.1"
  },
  "devDependencies": {
    "dotenv": "^17.2.2",
    "nodemon": "^3.1.10"
  }
}
```

Arquitectura de carpetas y archivos



Creación de Frontend

```
PS C:\Users\AizenLM\desktop\red-social> npm create vite@latest
Need to install the following packages:
create-vite@8.0.1
Ok to proceed? (y) y
> npx
> create-vite

|
◇ Project name:
| static
```

```
|
◇ Select a framework:
| React
|
◇ Select a variant:
| JavaScript
|
◇ Use rolldown-vite (Experimental)?:
| No
|
◇ Install with npm and start now?
| Yes
|
◇ Scaffolding project in C:\Users\AizenLM\desktop\red-social\static...
|
◇ Installing dependencies with npm...

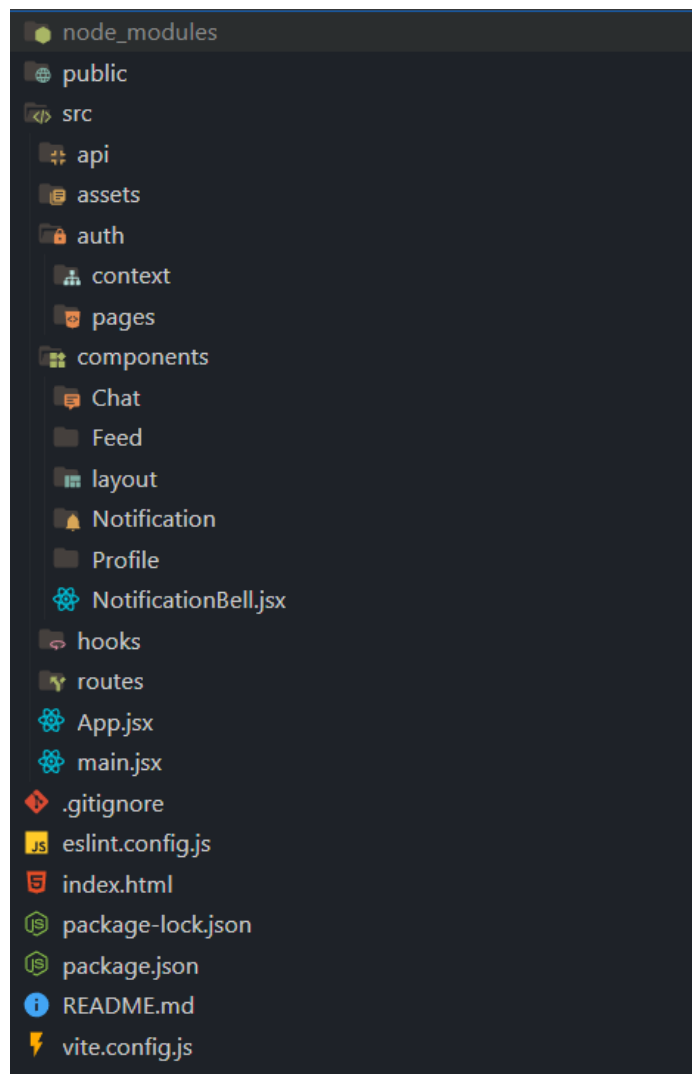
added 199 packages, and audited 200 packages in 16s
32 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```

Intalacion de dependencias

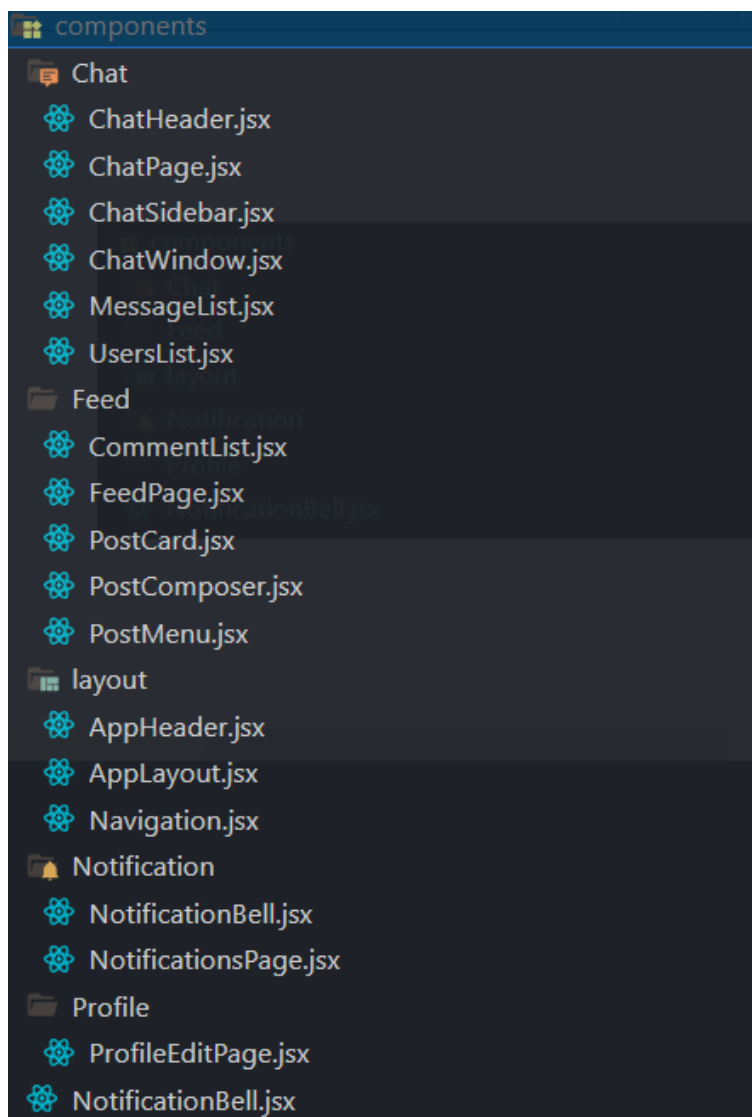
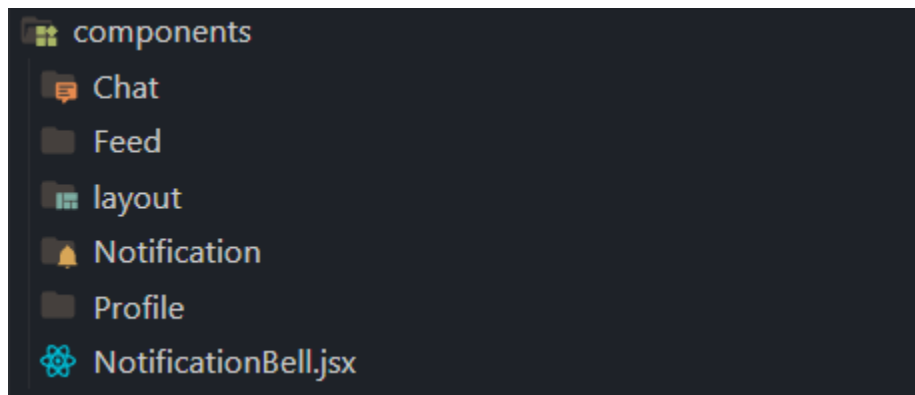
```
{
  "name": "01-static",
  "private": true,
  "version": "0.0.0",
  "type": "module",
  "scripts": {
    "dev": "vite",
    "build": "vite build",
    "lint": "eslint .",
    "preview": "vite preview"
  },
  "dependencies": {
    "@tailwindcss/vite": "^4.1.13",
    "axios": "^1.12.2",
    "react": "^19.1.1",
    "react-dom": "^19.1.1",
    "react-icons": "^5.5.0",
    "react-router-dom": "^7.9.1",
    "socket.io-client": "^4.8.1",
    "tailwindcss": "^4.1.13"
  },
  "devDependencies": {
```

```
"@eslint/js": "^9.36.0",
"@types/react": "^19.1.13",
"@types/react-dom": "^19.1.9",
"@vitejs/plugin-react": "^5.0.3",
"eslint": "^9.36.0",
"eslint-plugin-react-hooks": "^5.2.0",
"eslint-plugin-react-refresh": "^0.4.20",
"globals": "^16.4.0",
"vite": "^7.1.7"
}
}
```

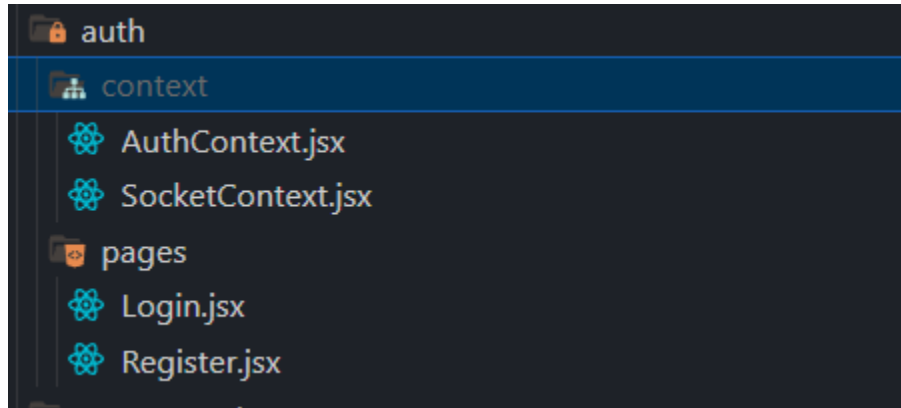
Arquitectura de carpetas



Componentes



Sistema de autenticación



Router.jsx

```
import { createBrowserRouter } from "react-router-dom";
import PrivateRoute from "../PrivateRoute";
import PublicRoute from "../PublicRoute";

// Pages
import Register from "../auth/pages/Register";
import Login from "../auth/pages/Login";

import App from "../App";

export const router = createBrowserRouter([
  {
    path: "/",
    element: (
      <PrivateRoute>
        <App></App>
      </PrivateRoute>
    ),
  },
  {
    path: "/login",
    element: (
      <PublicRoute>
        <Login />
      </PublicRoute>
    ),
  },
  {
    path: "/register",
    element: (
```



```

    <PublicRoute>
      <Register />
    </PublicRoute>
  ),
},
]);

```

App.jsx

```

import React, { useState } from 'react';
import { useAuth } from '../auth/context/AuthContext';

import { ChatPage } from '../components/Chat/ChatPage';
import { FeedPage } from '../components/Feed/FeedPage';
import { ProfileEditPage } from '../components/Profile/ProfileEditPage';
import { NotificationsPage } from
'../components/Notification/NotificationsPage';
import { AppLayout } from '../components/layout/AppLayout';

function App() {
  const { user } = useAuth();
  const [activePage, setActivePage] = useState('feed');
  const renderPage = () => {
    switch (activePage) {
      case 'feed':
        return <FeedPage />;
      case 'chat':
        return <ChatPage />;
      case 'profile':
        return <ProfileEditPage />;
      case 'notifications':
        return <NotificationsPage />;
      default:
        return <FeedPage />;
    }
  };
  return (
    <AppLayout activePage={activePage} onPageChange={setActivePage}>
      {renderPage()}
    </AppLayout>
  );
}

export default App;

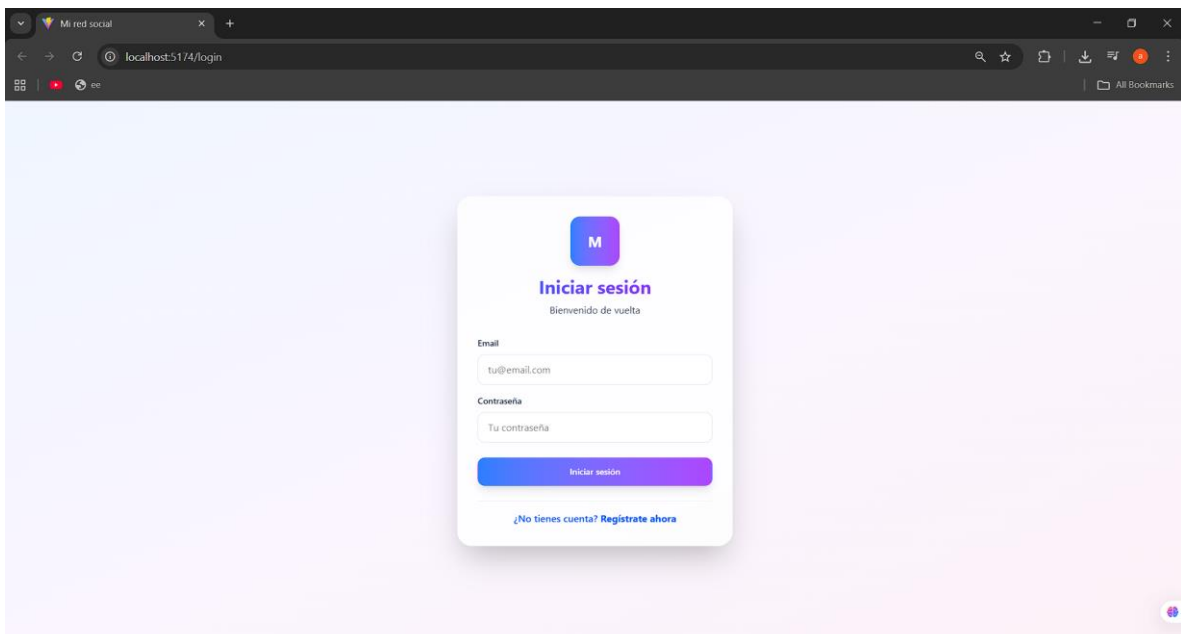
```

Main.jsx

```
import { StrictMode } from "react";
import { createRoot } from "react-dom/client";
import { RouterProvider } from "react-router-dom";
import { router } from "../routes/router";
import "../src/assets/styles/main.css";
import { AuthProvider } from "../auth/context/AuthContext";
import { SocketProvider } from "../auth/context/SocketContext";
createRoot(document.getElementById("root")).render(
  <StrictMode>
    <AuthProvider>
      <SocketProvider>
        <>
          <RouterProvider router={router}></RouterProvider>
        </>
      </SocketProvider>
    </AuthProvider>
  </StrictMode>
);
```

Resultado final

Páginas de iniciar sesión y registro



mi red social

localhost:5174/register

M

Crear cuenta

Únete a nuestra comunidad

Nombre de usuario *

Ubicación

Email *

Intereses (separados por comas)

Contraseña *

Confirmar *

Biografía

Teléfono

Edad

Crear cuenta

[¿Ya tienes cuenta? Inicia sesión](#)

mi red social

localhost:5174/register

M

Crear cuenta

Únete a nuestra comunidad

Nombre de usuario *

Ubicación

Email *

Intereses (separados por comas)

Contraseña *

Confirmar *

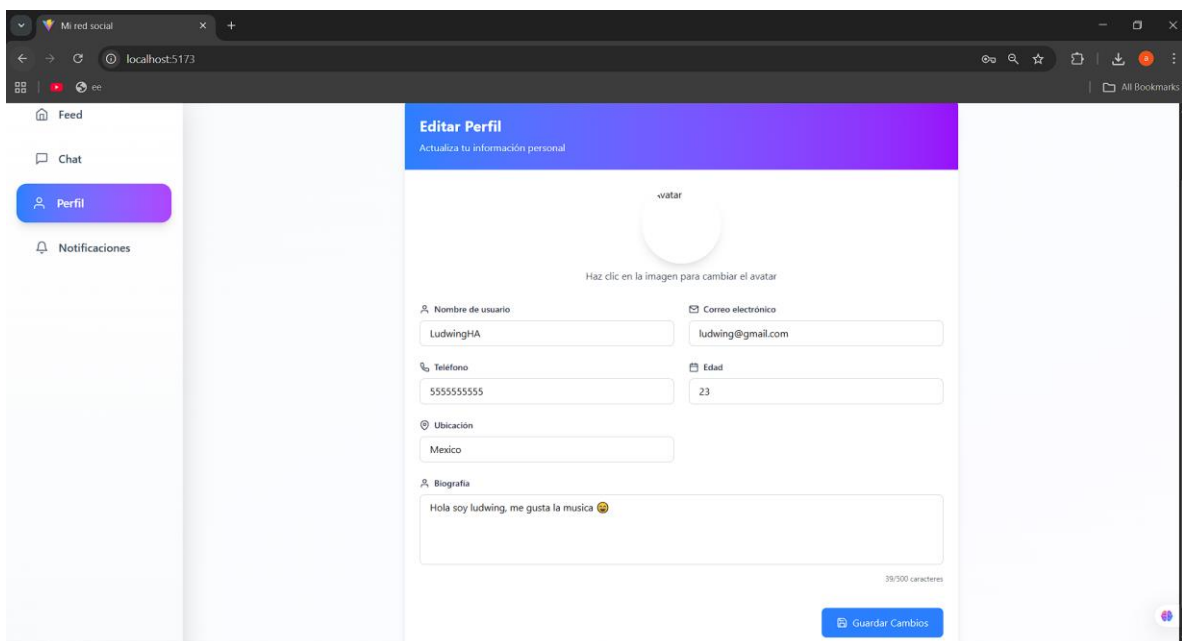
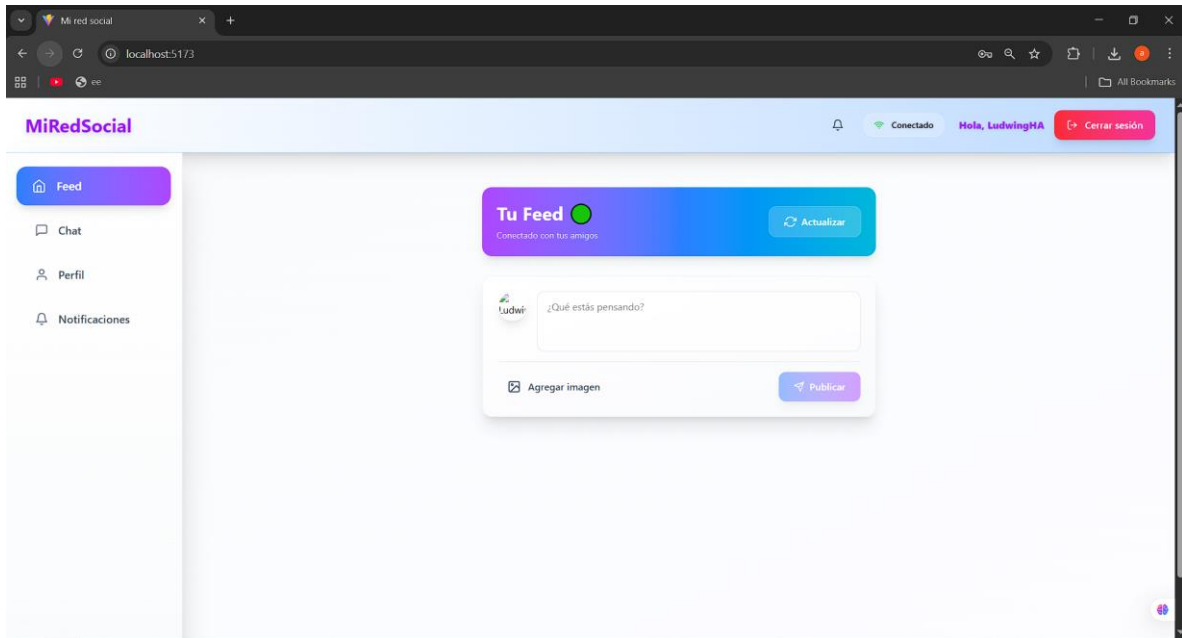
Biografía

Teléfono

Edad

Crear cuenta

[¿Ya tienes cuenta? Inicia sesión](#)



Editar Perfil

Actualiza tu información personal



Haz clic en la imagen para cambiar el avatar

Nombre de usuario

LudwingHA

Correo electrónico

ludwing@gmail.com

Teléfono

555555555

Edad

23

Ubicación

Mexico

Biografía

Hola soy ludwing, me gusta la musica 🎵

39/500 caracteres

Guardar Cambios

Feed

Chat

Perfil

Notificaciones

Tu Feed

Conectado con tus amigos

Actualizar

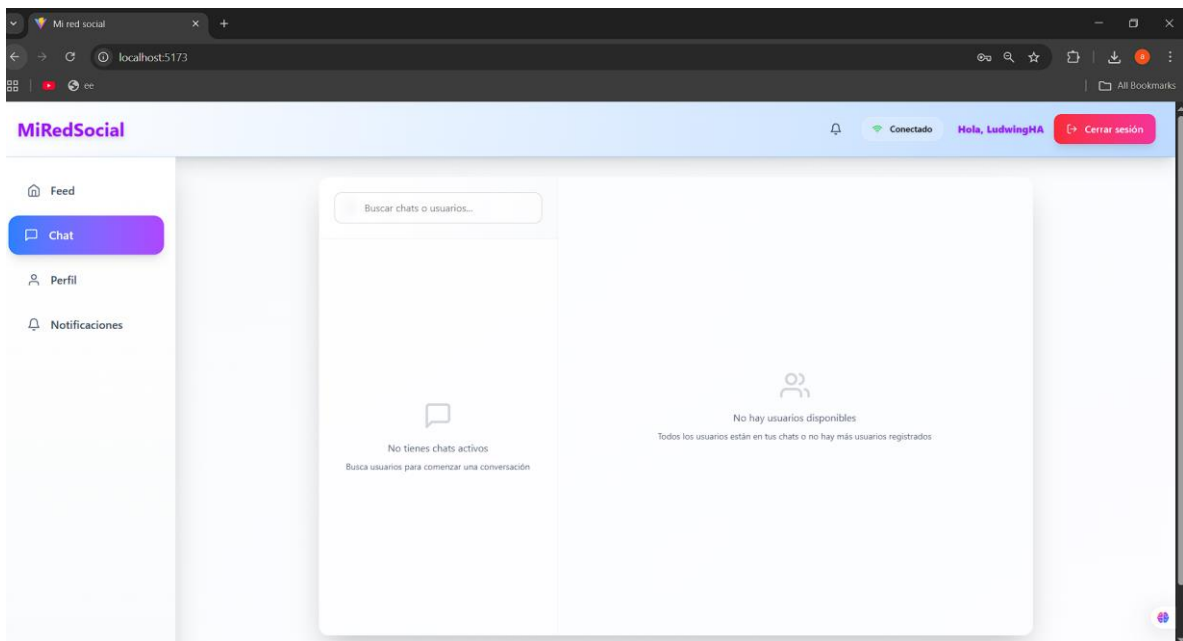
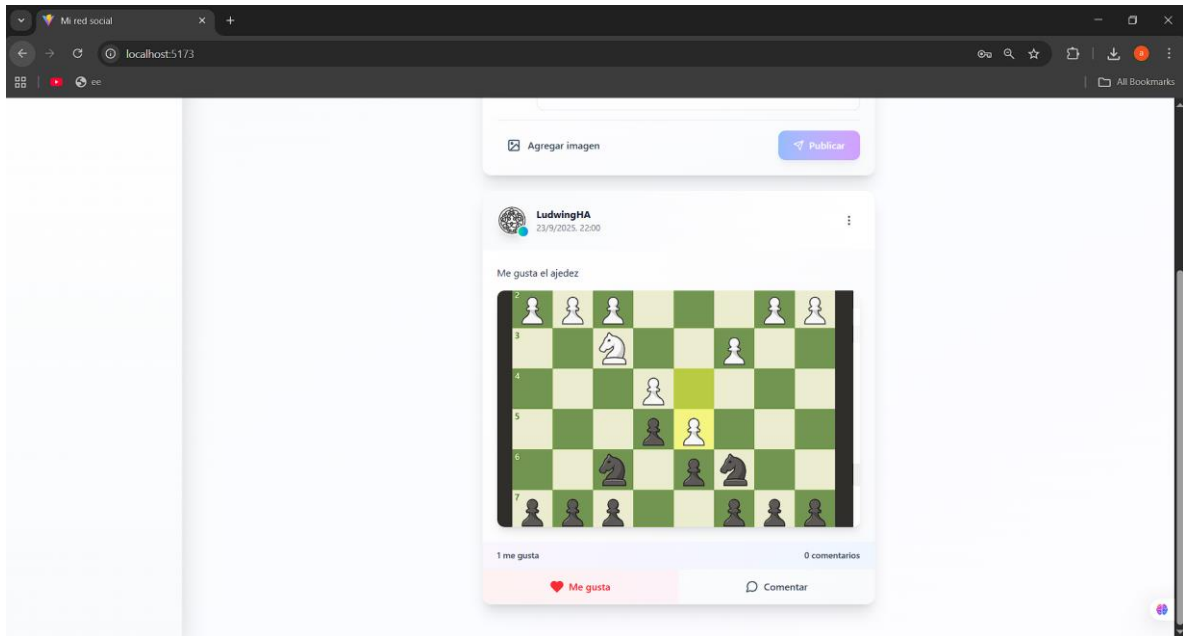


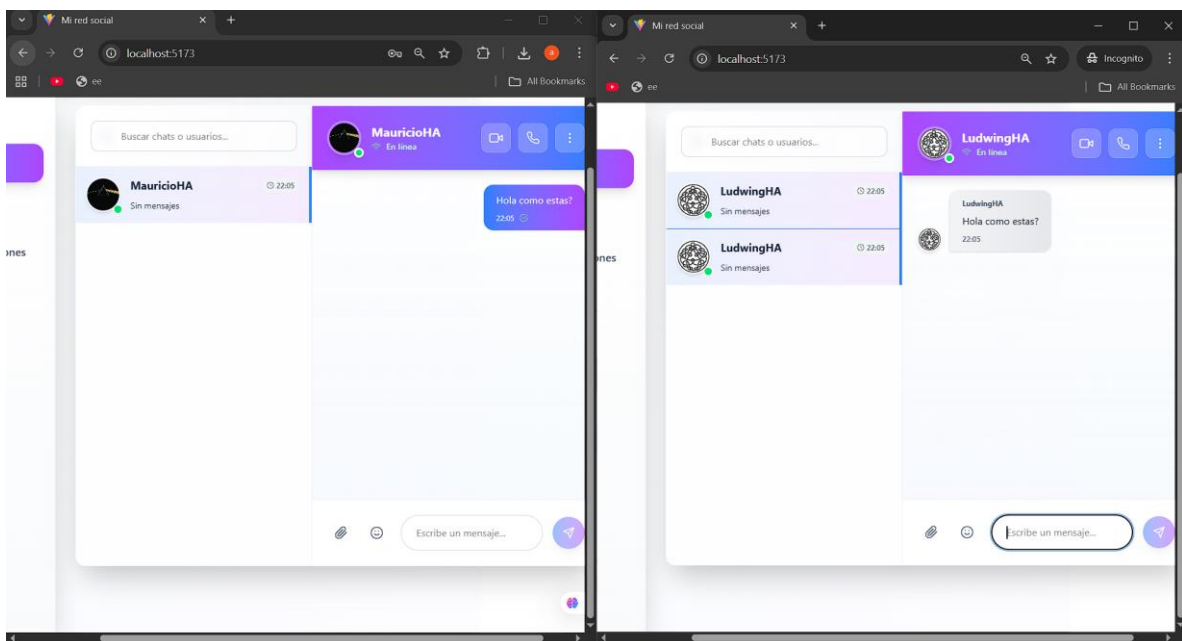
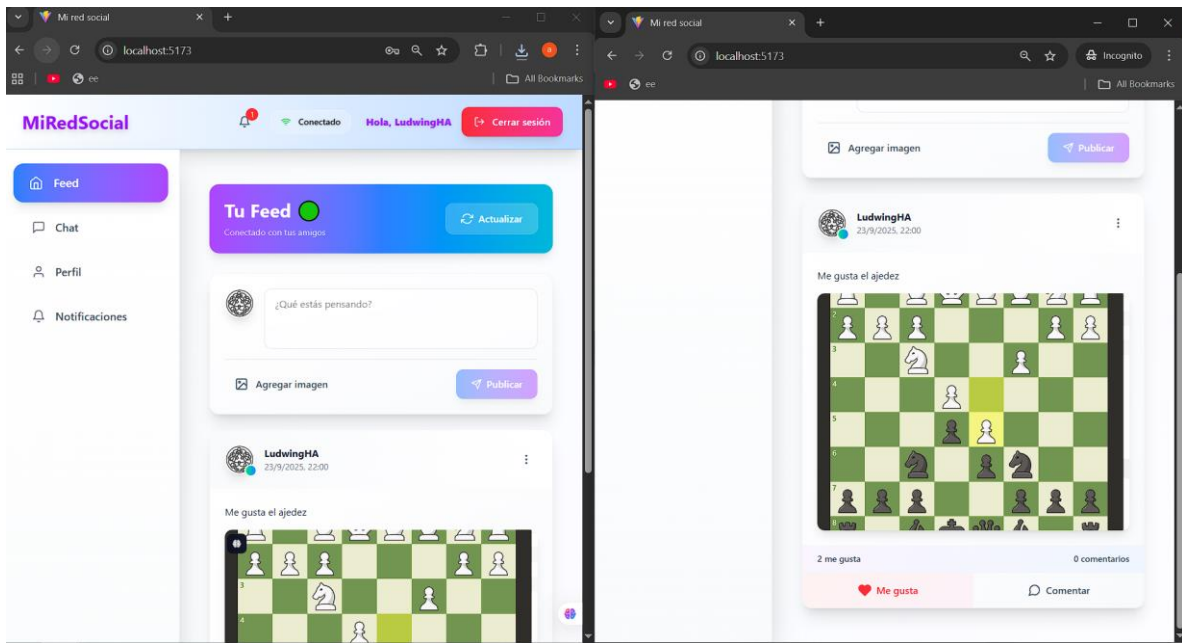
Me gusta el ajedrez

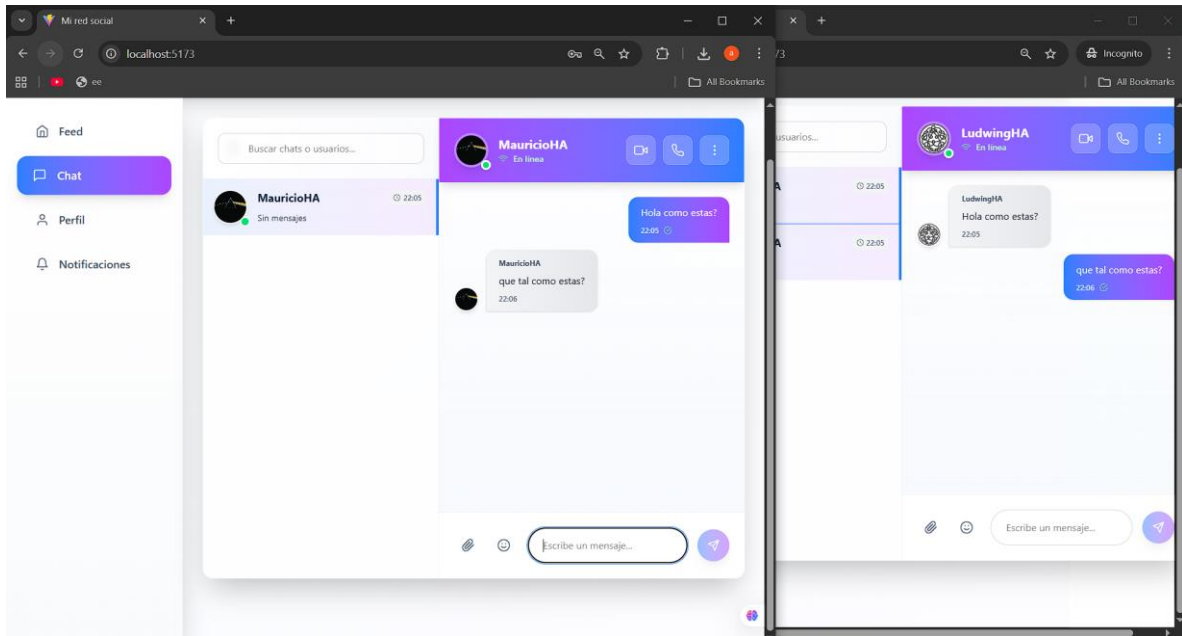


Agregar imagen

Publicar







Conclusión

Al completar este proyecto, logramos comprender a profundidad cómo funciona una aplicación web moderna de pila completa. Aprendimos a crear una arquitectura donde el frontend, construido con React.js, es responsable de renderizar el contenido directamente en el navegador del usuario (CSR). Esta estrategia nos permitió crear una experiencia de usuario rápida y fluida, con transiciones de página instantáneas.

Además, consolidamos el uso de un backend robusto con Express, que gestiona la lógica de la aplicación y se comunica con una base de datos MongoDB para almacenar los datos. La integración de Socket.IO fue crucial para implementar funciones de tiempo real, como la mensajería instantánea, demostrando cómo la comunicación bidireccional entre el cliente y el servidor enriquece la interactividad de una aplicación.

En resumen, este proyecto sirvió como una experiencia práctica completa para entender la interacción entre diferentes tecnologías. Nos mostró cómo la elección de una estrategia de renderizado como CSR influye en la arquitectura, y cómo herramientas específicas se combinan para construir una aplicación web dinámica, eficiente y funcional.