

UNIVERSIDAD COMUNERA



TALLER IV

Profe Guia: Jhony Benítez

INTEGRANTES:

- 1. MAIHARA RODRIGUEZ**
- 2. JULIO DANIERI**

2023

INFORME DEL PROYECTO

El proyecto actual se desarrolla en colaboración con Teach For All y tiene como objetivo principal optimizar el control de información de los estudiantes, especialmente en lo que respecta a asistencias y otros datos relevantes manejados por los profesores. Hasta el momento, este proceso se realiza manualmente mediante hojas de cálculo en Excel. Con la intención de mejorar la eficiencia y la gestión de datos, nuestro equipo, compuesto por Julio Danieri y Maihara Rodriguez, ha decidido implementar un sistema utilizando tecnologías web modernas, específicamente React para el frontend, Node.js para el backend, y PostgreSQL como base de datos.

Fase de Aprendizaje:

Ambos miembros del equipo han pasado aproximadamente tres semanas dedicadas a adquirir conocimientos inicialmente sobre React y Node.js. Este periodo de aprendizaje incluyó la participación en cursos de youtube en línea, la lectura de documentación oficial y la práctica mediante ejemplos simples. La elección de estas tecnologías se basa en su versatilidad, popularidad y la capacidad de crear aplicaciones web eficientes y dinámicas.

Una vez adquiridos los conocimientos necesarios, el equipo inició el desarrollo del proyecto desde cero. Las primeras etapas del proyecto se centraron en la implementación de las funcionalidades básicas de autenticación, como iniciar sesión y registrarse en la plataforma. Esto implica la creación de páginas y formularios correspondientes tanto en el frontend como en el backend.

Hasta Ahora:

Hasta la fecha, el proyecto ha alcanzado un hito importante al permitir a los usuarios iniciar sesión y registrarse en la plataforma. Se han establecido las bases para la gestión de usuarios y la autenticación, sentando así las bases para las futuras funcionalidades del sistema. La interfaz de usuario se ha diseñado con la usabilidad en mente, proporcionando una experiencia intuitiva y fácil para los usuarios finales.

A continuación se explica que se tiene hasta ahora que es para iniciar sesión y registrarse en la página.

Servidor Backend (server.js):

- Express y Pool configura el servidor Express y la conexión a PostgreSQL utilizando el módulo `express` y `pg.Pool`.
- Bcrypt y Cors importa las bibliotecas `bcrypt` para el cifrado de contraseñas y `cors` para habilitar el manejo de solicitudes entre dominios diferentes.
- Se tiene endpoints para registro e Inicio de Sesión, en donde se define las rutas `/signup` y `/login` que manejan las solicitudes de registro e inicio de

sesión respectivamente. Utiliza la base de datos PostgreSQL para almacenar y verificar la información.

- Un manejo de errores que implementa para devolver respuestas adecuadas en caso de fallos durante las consultas a la base de datos.

Frontend (App.js):

Se utiliza React Router para gestionar las rutas de la aplicación. Importa y utiliza componentes como Navegación, Signup, Login, Welcome, infousuario y Pantalla.

Navegación (navegacion.js):

Se diseña un encabezado de navegación con el nombre "Teach For All". Incluye botones para registrarse e iniciar sesión, con estilos específicos.

Componentes de Páginas (signup.js, login.js, welcome.js, pantalla.js, infousuario.js):

- El formulario de registro captura nombre, correo electrónico, contraseña y rol(profesor, exp o director).
- Se envía datos al backend en donde almacena esta información en la base de datos PostgreSQL.
- El formulario de inicio de sesión envía las credenciales al backend para verificar la autenticidad y proporciona un mensaje de alerta en caso de éxito o fallo.
- La página de bienvenida muestra el rol del usuario y contiene un botón para redirigir a una nueva pantalla.
- La pantalla es un componente de página simple que muestra un mensaje de bienvenida.
- La pantalla de la información de usuario recopila todos los datos guardados en la base de datos y que tiene visible el usuario una vez inicie sesión en la página.

Estilos y Diseño

Se utilizan estilos en línea en algunos componentes para personalizar la apariencia, como colores de fondo y estilos de botones.

Stack Tecnológico Utilizado

Frontend

- React: Un framework de JavaScript para construir interfaces de usuario interactivas y eficientes.
- CSS: Utilizado para estilizar y dar formato a la interfaz de usuario.
- Bootstrap: Un marco de diseño que facilita la creación de interfaces de usuario atractivas y responsivas.

Backend

- Node.js: Un entorno de ejecución de JavaScript en el lado del servidor que permite construir aplicaciones escalables y de alto rendimiento.
- Express: Un framework web para Node.js que simplifica la creación de API RESTful y la gestión de rutas.
- PostgreSQL: Un sistema de gestión de bases de datos relacional robusto y escalable.

Herramientas Adicionales:

- npx create-react-app: Utilizado para iniciar rápidamente un proyecto React con una estructura y configuración predefinidas.
- Cors: Un middleware para Express que facilita la gestión de las políticas de mismo origen (CORS).
- Nodemon: Utilizado para reiniciar automáticamente el servidor Node.js cuando se realizan cambios en el código.

Complicaciones y Desafíos Encontrados en el Proyecto:

- **Problemas con GitHub en el Frontend:** Se encontró un error peculiar en GitHub relacionado con el frontend, donde los archivos parecían comprimidos y no se podían abrir correctamente. Este problema afectó la colaboración y el flujo de trabajo del equipo estancándolo al inicio.
- **Desafíos durante el Período de Aprendizaje:** Iniciar el proyecto con tecnologías completamente nuevas, como React, Node.js y PostgreSQL, representó un desafío significativo. El período de aprendizaje desde cero implicó la necesidad de realizar cursos en línea y leer documentación para familiarizarse con estos lenguajes y herramientas.
- **Problemas con las Computadoras de los Integrantes:** Las computadoras de los integrantes del equipo sufrieron problemas técnicos durante el desarrollo del proyecto. Uno de los integrantes se quedó sin espacio en disco, lo que afectó el rendimiento. Otro enfrentó problemas donde su computadora dejó de responder, complicando aún más el avance del proyecto.
- **Perdida de Integrantes en el Grupo:** La pérdida de integrantes en el grupo fue un desafío, ya que disminuyó la capacidad de trabajo y colaboración. La falta de recursos humanos afectó la distribución de tareas y la velocidad de desarrollo.
- **Falta de Tiempo por Compromisos Laborales:** Compromisos laborales de ambos integrantes del equipo limitaron el tiempo disponible para el proyecto. La gestión del tiempo se volvió crítica y afectó el progreso del desarrollo.
- **Trabajo sin una Guía Documentada:** La falta de una guía documentada o un documento de especificaciones claras dificultó la estructuración y

planificación del proyecto. Esto llevó a cierta incertidumbre en la dirección y la implementación de características específicas.

- **Problemas en la Base de Datos:** Se experimentaron problemas en la base de datos, especialmente en el proceso de inicio de sesión. La base de datos no respondía correctamente en algunos casos o no almacenaba datos de manera consistente, lo que afectaba la funcionalidad central del sistema.

Conclusión

El proyecto ha logrado implementar funcionalidades clave, como registro e inicio de sesión, así como la estructura básica de navegación. Se pone un énfasis significativo en la usabilidad y el diseño atractivo. El siguiente paso debería centrarse en expandir las funcionalidades para abordar específicamente las necesidades de Teach For All en el control de información de los estudiantes.

LINK DE ACCESO AL GITHUB DEL PROYECTO FINAL

- <https://github.com/maihaara/tallerIV>