

Planificación de actividades para la implementación de software



Integrantes:

Luisa Campos

Lizeth Duitama

Robinson Chaguala

Michael Quecano

Ficha: 2902290

Introducción

El presente proyecto tiene como objetivo el desarrollo e implementación de un sistema integral de gestión de novedades para el Servicio Nacional de Aprendizaje (SENA), que permitirá gestionar de manera eficiente y centralizada las novedades académicas, disciplinarias y de ambiente de la comunidad SENA. Esta herramienta será de gran utilidad para los distintos actores que hacen parte de esta institución, tales como aprendices, coordinadores, instructores y administradores de edificios, facilitando la comunicación y el seguimiento oportuno de los procesos relacionados con la vida académica y administrativa de los estudiantes.

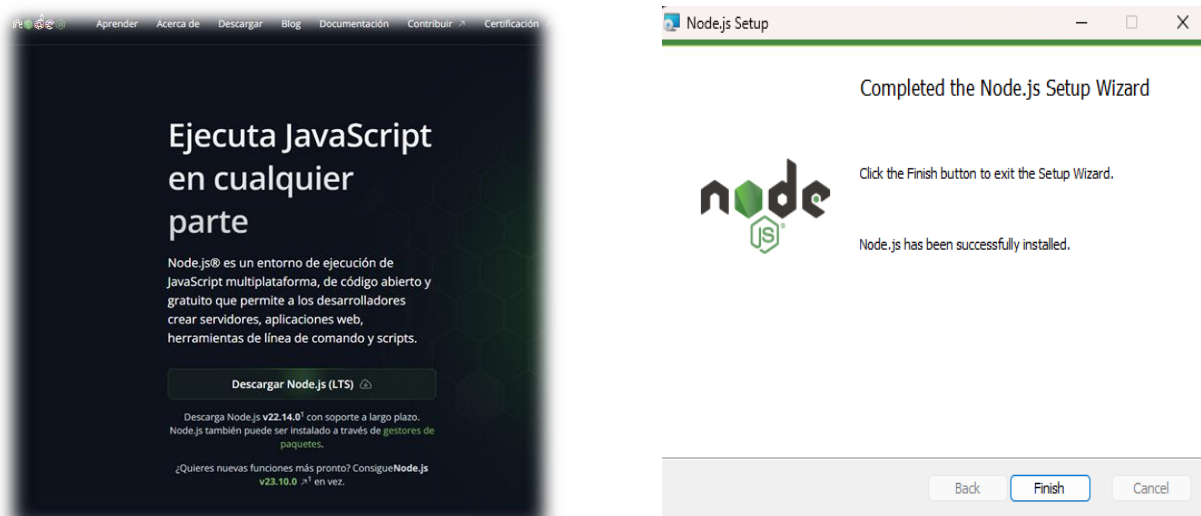
A través de este sistema, se podrán registrar, consultar y gestionar incidencias tanto académicas como disciplinarias, además de facilitar los procesos de decisión para el retiro voluntario de los aprendices o la desvinculación de los mismos. También se brindará soporte en el manejo de situaciones relacionadas con el ambiente y la infraestructura, contribuyendo a un mejor control de las condiciones que afectan el desempeño y bienestar de los aprendices.

Este sistema tiene como finalidad optimizar los recursos humanos y tecnológicos del SENA, mejorar la toma de decisiones, reducir tiempos de respuesta y garantizar la transparencia y trazabilidad en todos los procesos. De esta forma, se busca una gestión más eficiente, que favorezca tanto el desarrollo académico de los aprendices como el bienestar de toda la comunidad educativa.

Preparación de la plataforma

Instalar Node.js

Si no tienes **Node.js** instalado en tu sistema, puedes descargarlo e instalarlo desde su [sitio web oficial](#) Expo requiere Node.js para funcionar correctamente.



Instalar Expo CLI

Expo CLI es una herramienta de línea de comandos que te permite crear, construir y administrar aplicaciones con Expo.

Abre la terminal y ejecuta el siguiente comando para instalar Expo CLI de manera global:

```
C:\Users\Aprendiz>npm install -g expo-cli
```

```
added 1305 packages in 2m

189 packages are looking for funding
  run `npm fund` for details
npm notice
npm notice New major version of npm available! 10.9.2 -> 11.2.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.2.0
npm notice To update run: npm install -g npm@11.2.0
npm notice
```

Crear un nuevo proyecto con Expo

Una vez que Expo CLI esté instalado, puedes crear un nuevo proyecto de React Native utilizando Expo con el siguiente comando:

```
C:\Users\Aprendiz>expo init GestionNovedades
```

Navegar al directorio del proyecto

Una vez creado el proyecto, navega al directorio del proyecto:

```
C:\Users\Aprendiz>cd GestionNovedades
```

Iniciar el servidor de desarrollo

Para empezar a desarrollar tu aplicación, inicia el servidor de desarrollo de Expo con el siguiente comando:

```
C:\Users\Aprendiz\GestionNovedades>npm start
```

```
Starting project at C:\Users\Aprendiz\GestionNovedades
Starting Metro Bundler
```



```
> Metro waiting on exp://172.30.7.14:8081
> Scan the QR code above with Expo Go (Android) or the Camera app (iOS)

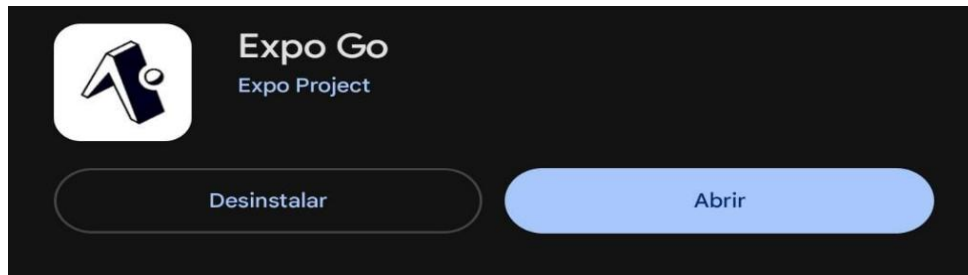
> Using Expo Go
> Press s | switch to development build

> Press a | open Android
> Press w | open web

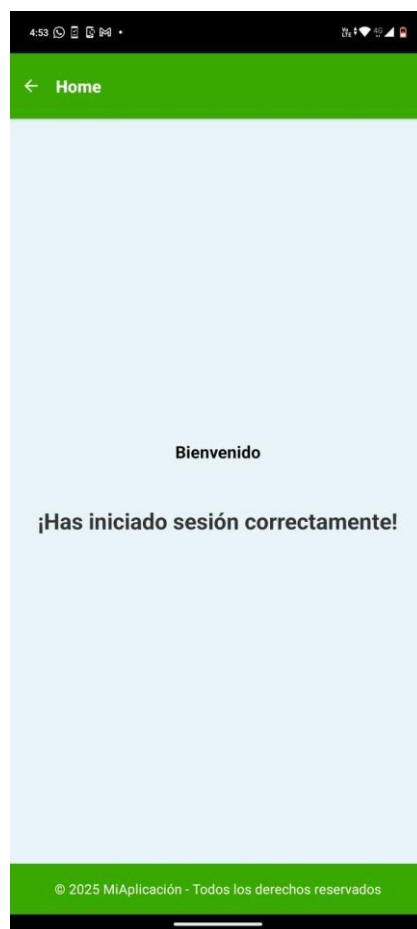
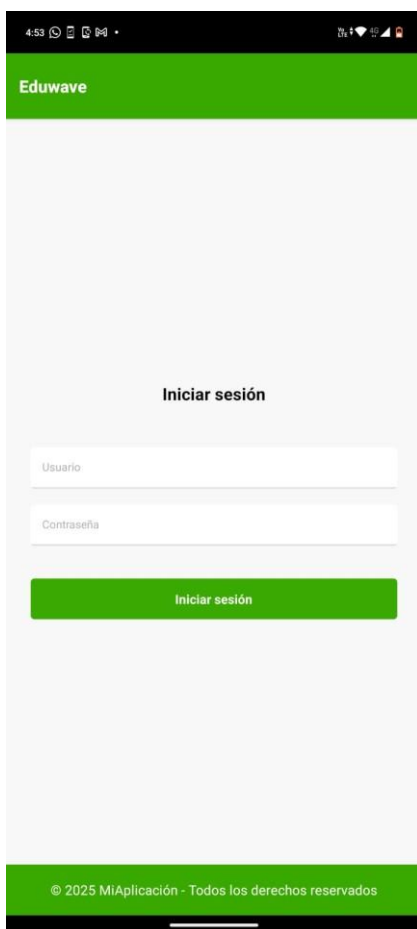
> Press j | open debugger
> Press r | reload app
> Press m | toggle menu
```

Instalar Expo Go

Es una aplicación y entorno de pruebas que permite crear aplicaciones para iOS y Android.



Resultado



Navegacion

`npm install @react-navigation/native`

Este paquete es el núcleo de la biblioteca de navegación en React Native. Proporciona la infraestructura principal que permite manejar la navegación entre pantallas dentro de la aplicación. `@react-navigation/native` te permite usar distintos tipos de navegación, como pilas (stack), pestañas (tabs) o cajones (drawer).

`npm install @react-navigation/stack`

Este paquete es una extensión de `@react-navigation/native` y te permite usar una navegación basada en pilas (stack navigation). Es decir, maneja una serie de pantallas apiladas que el usuario puede ir "apilando" y navegar hacia adelante o hacia atrás.

`npm install react-native-screens react-native-safe-area-context`

Estos paquetes son necesarios para optimizar la navegación en React Native.

- *react-native-screens*: Este paquete mejora el rendimiento de la navegación al hacer que las pantallas se monten y desmonten de manera más eficiente. Ayuda a que la navegación sea más fluida, utilizando un enfoque más eficiente para gestionar las vistas.
- *react-native-safe-area-context*: Este paquete es útil para manejar correctamente las áreas seguras en los dispositivos (como los bordes curvados de las pantallas o la muesca del teléfono) asegurando que el contenido no se superponga a estas áreas. Es esencial para hacer aplicaciones que se vean bien en todos los dispositivos.

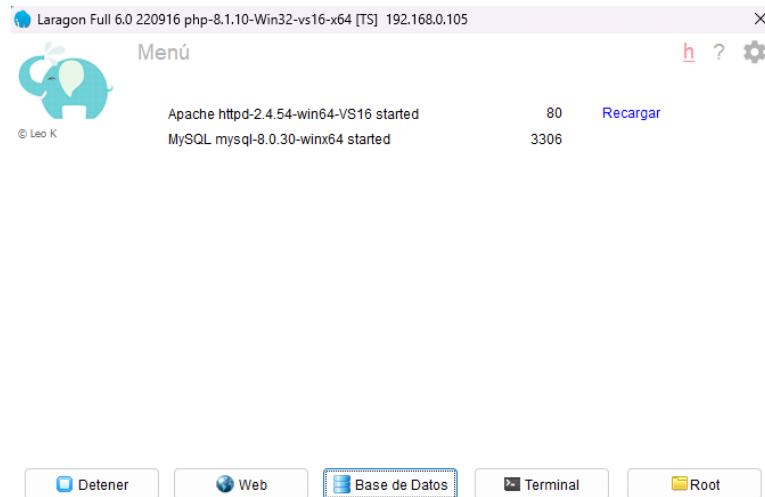
`npm install react-native-gesture-handler`

Este paquete permite gestionar gestos táctiles, como deslizamientos, toques y desplazamientos. Es esencial para habilitar gestos interactivos en la navegación, como deslizar para volver a la pantalla anterior (en la navegación tipo pila) o para gestionar otros gestos como el deslizamiento en pestañas o menús. Sin este paquete, muchas de las interacciones de navegación basadas en gestos no funcionarían correctamente.

CONEXIÓN BASE DE DATOS

1. Instala Laragon y asegúrate que MySQL esté corriendo Abre Laragon.

Verifica que los servicios estén "Running" (especialmente MySQL y Apache, si los usas).
Anota el puerto de MySQL (normalmente 3306).

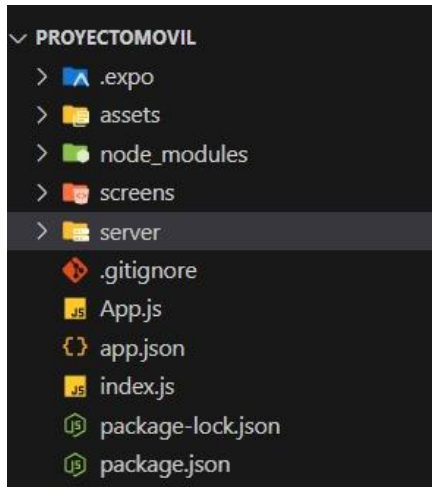


2. Crea una base de datos y tabla de usuario

En phpMyAdmin (en <http://localhost/phpmyadmin>), ejecuta algo como:



3. Crea una nueva carpeta en la ruta del proyecto que se llame server



4. Dentro de esa carpeta ejecuta: npm init -y

```
PS C:\Users\MIKE\ProyectoMovil\server> npm init -y
```

Esto crea un archivo package.json



5. Instala las dependencias necesarias:

- npm install express mysql cors body-parser

```
PS C:\Users\MIKE\ProyectoMovil\server> npm install express mysql cors body-parser
```


Esto instala:

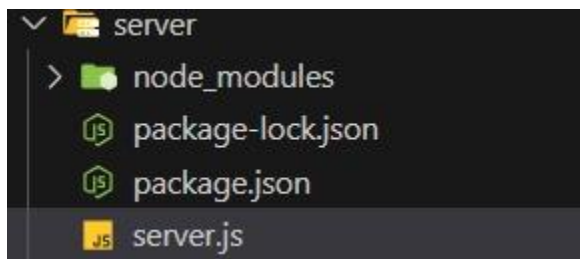
express: servidor web

mysql: para conectarte a MySQL

cors: para que tu app Expo pueda comunicarse

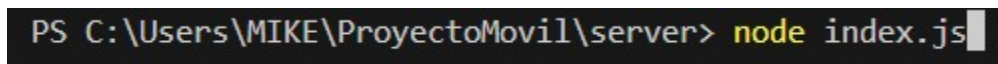
body-parser: para leer datos del body de las peticiones

6. Despues creamos el archivo server.js dentro de nuestra carpeta server

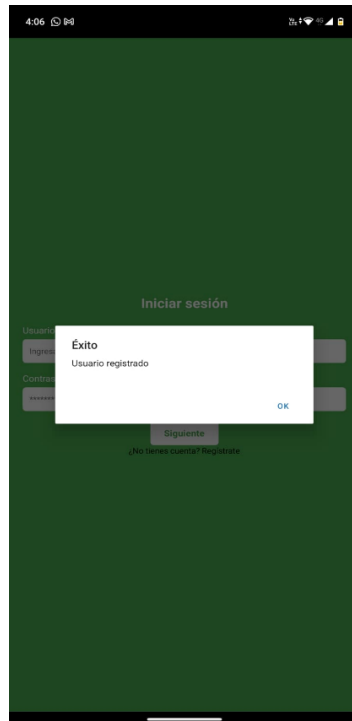
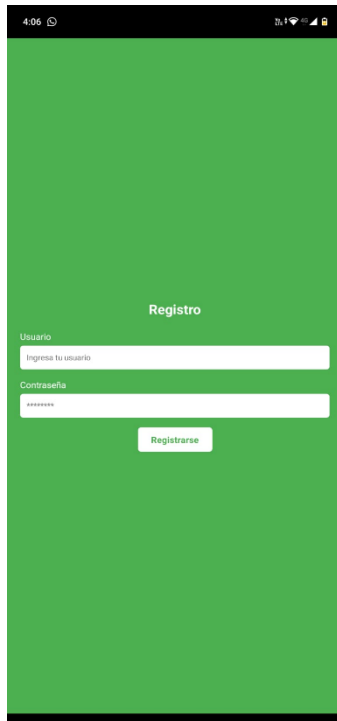


7. Iniciar el servidor

node index.js



RESULTADO



usuario	contraseña
Juan	2007
Liz	1234

