

# Diseño y Programación de Software

## Multiplataforma

Universidad Don Bosco  
Campus Soyapango  
2024

# PORTAL ACADÉMICO

05 de mayo del 2024

## MANUAL TÉCNICO SOBRE NUESTRO PROYECTO

En el presente manual, se llevará a cabo el uso adecuado de las herramientas y tecnologías empleadas al momento de utilizar nuestro portal web, por lo que hemos abarcado el uso de ciertas metodologías, explicando a detalle cada paso empleado.

Tomando en cuenta esto, nuestro equipo se basó en una necesidad de cada estudiante, docente y personal administrativo, , llevando a cabo ciertas funciones que permiten al usuario poder ver, subir y hacer uso de un portal académico en cualquier centro educativo.

## Metas

1. **Optimizar la gestión académica:** Ofrecemos herramientas para la gestión de inicio de sesión, calificaciones, horarios de clases y algún otro aspecto para la administración académica .
2. **Facilitar el acceso a la información:** Proporciona un acceso fácil y rápido a recursos educativos, métodos y contenidos relevantes para los estudiantes, docentes y el personal administrativo.

## Especificaciones

La creación de un portal académico puede adaptarse según las necesidades y prioridades específicas de la institución educativa o la organización que lo esté desarrollando.

## Recursos utilizados.

### 1. Arquitectura del Sistema

Nuestro portal académico se basa en:

**Componentes:** Contiene el header con funciones para poder hacer un GoBack, muestra el usuario si es Docente o Estudiantes.

**Screens:** Contiene las vistas según el rol que es logrado, si es:

**Alumno:** muestra las vistas sobre horario de clases, calificaciones y actividades pendientes y/o agregadas.

**Docente:** muestra las vistas de horario de clases, calificaciones para alumnos, actividades agregadas o subir actividades.

**Registro:** Este puede agregar un usuario nuevo según el registro de datos almacenados.

**Login:** Puede loguearse según el rol asignado y redirigir a las vistas de estudiante o docente.

**Firebase:** Utilizamos Firebase para hacer uso de los almacenamientos de datos sobre el registro del usuario.

**Utils:** Tiene como contenido la tipografía, colores utilizados e imágenes, y datos tipo json, estos son utilizados para la estética de nuestro proyecto.

## 2. Tecnologías utilizadas.

Como principal tecnología empleada para nuestro proyecto está.

**React Native:** un framework JavaScript para crear aplicaciones reales nativas para iOS y Android. En este caso para hacer uso de nuestro proyecto.

**Firebase:** base de datos NoSQL alojada en la nube que te permite almacenar y sincronizar datos entre usuarios en tiempo real. Para nuestro proyecto utilizamos Firebase para hacer el registro de usuarios.

**Visual Studio Code:** editor de código fuente independiente que se ejecuta en Windows, macOS y Linux desde este editor de código inicializamos el proyecto.

**Snack Expo:** herramienta online para correr Expo en el browser, permite probar y compartir pequeñas aplicaciones o ejemplos, como alternativa para hacer la creación del proyecto.

**Node.js:** sirve para crear sitios web dinámicos muy eficientes, escritos con el lenguaje de programación JavaScript. Utilizando librerías de node.js hemos creado muchas de las interfaces visualizadas en el proyecto.

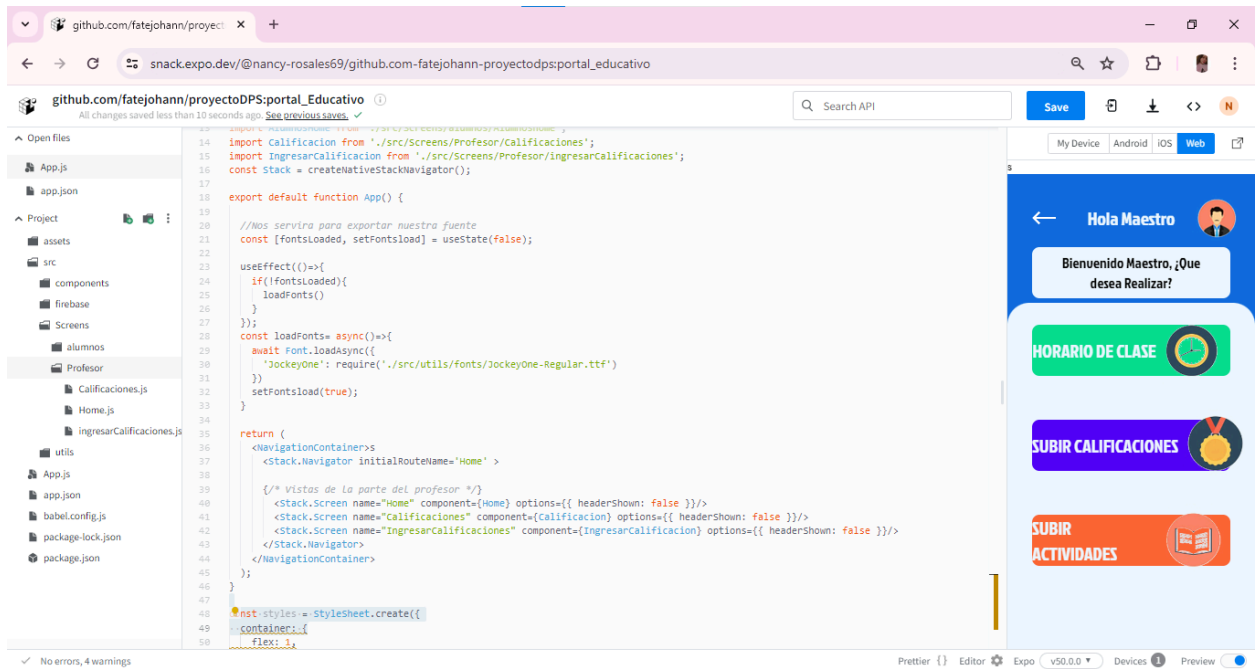
**gitHub:** Bajo nuestro repositorio, está el proyecto con todas los envíos del proyecto.

**Trello:** Actualizaciones y otros detalles.

**Figma:** Plataforma para hacer uso de nuestra idea para las interfaces del usuario.

## 3. Instalación y configuración

Al hacer uso del proyecto lo hemos enviado desde un repositorio en gitHub. Al descargar o exportar el proyecto dependiendo dónde será ejecutado el proyecto si en Snack Expo dev, deberá traer el proyecto desde github y descargar las dependencias. Luego si desea observar el proyecto desde su dispositivo móvil, dispositivo web o en la web, en el lado derecho entrá esas opciones



De otro modo si es exportado a Visual Studio Code, desde la terminal deberá ejecutar el siguiente comando y tomar una espera para que termine de descargar las dependencias, posterior a eso se deberá correr el proyecto ya sea desde el comando “`npx expo start`” o “`npm run android`”.

```
PS C:\Users\v3512\OneDrive\Desktop\proyectoDPS-main> npm i
```

```
PS C:\Users\v3512\OneDrive\Desktop\proyectoDPS-main> npm run android
```

```
PS C:\Users\v3512\OneDrive\Desktop\proyectoDPS-main> npx expo start
```

#### 4. Funcionalidades principales:

Como principal funcionalidad, dependiendo del usuario:

- Funcionalidades para Estudiantes:
  - Visualización de Notas: Se explica cómo acceder y visualizar las calificaciones de las diferentes materias.
  - Consulta de Horarios: Se detalla cómo consultar y acceder al horario de clases personalizado.
- Funcionalidades para Profesores
  - Carga de Notas: Instrucciones para ingresar y actualizar las calificaciones de los estudiantes de manera eficiente.
  - Gestión de Horarios: Cómo establecer y modificar los horarios de clase de manera conveniente.

## 5. Resultados de pruebas:





## 6. Mantenimiento y escalabilidad:

- Como uno de los mantenimientos para este será ir revisando cualquier bug o alguna funcionalidad que no esté dándose como debería.
- Por otro lado, como un método de escalabilidad, se irán agregando más funcionalidades para el uso cotidiano a nuestro portal académico.

## 7. Referencias:

Librerías utilizadas:

```
npm install @react-navigation/native-stack
```

```
npm install @react-navigation/stack
```

```
npm i firebase
```

```
npm install --save @react-native-firebase/app @react-native-firebase/firestore
```

```
npm install --save react-native-snap-carousel
```