

TRABAJO FINAL

DESARROLLAR Y PRESENTAR:

- Proyecto Frontend alojado en repositorio [Github](#) a nombre del alumno.
- Proyecto Backend alojado en repositorio [Github](#) a nombre del alumno.
- Ambos repositorios deben contener código prolijo, segmentado en commits.
- Ambos proyectos deben estar hosteados en la nube por separado, cada uno debe poder accederse a través de una url y deben comunicarse entre sí a través del protocolo HTTP.
- Base de datos hosteada en la nube, la cual será consumida por el proyecto backend.

(Se puede utilizar cualquier tecnología siempre y cuando se respete la arquitectura [API REST](#))

FUNCIONALIDAD MÍNIMA REQUERIDA:

- Ruta pública en el Frontend que visualice datos traídos desde el backend.
- Ruta pública de login en el Frontend que permita ingresar usuario y contraseña para iniciar sesión (con sus respectivas validaciones).
- Ruta privada en el Frontend (solo se puede acceder si se inicia sesión) con un CRUD* de datos guardados en la base de datos (con sus respectivas validaciones).
- El CRUD* debe afectar de forma directa los datos que se muestran en la pantalla pública.
- Los endpoints utilizados para el CRUD deben ser privados, es decir, solo se pueden realizar si el usuario está autenticado a través de un token válido.
- Funcionalidad de Logout en el Frontend y redirección al Home.

Tecnologías recomendadas:

- Tecnologías frontend: Flexbox, React, React Router y React Hook Form.
- Tecnologías backend: Node.js, Express, MongoDB, Mongoose y Firebase.
- Cloud Server: Heroku / Vercel.

El proceso de evaluación del examen final será:

1. Abrir la aplicación web cliente en Google Chrome.
2. Visitar la página pública y verificar que los datos que se muestran están en la base de datos.
3. Acceder a la pantalla de login, verificando el manejo de errores, y al loguearse que redirija a la página privada.
4. Desde la página privada, ver el listado completo, dar de alta, eliminar y modificar los datos, y verificar que se actualizan en la base de datos.
5. Al eliminar se debe visualizar un cartel de confirmación, que permite cancelar o confirmar la operación.
6. Hacer logout y que se redirija a la pagina publica.
7. Ingresar por URL a la página privada estando deslogueado, y verificar que no permita acceder, o que redirija a la pantalla de login.
8. Revisar la calidad del código del Backend y del Frontend en Github.
9. Revisar el correcto entendimiento de las funcionalidades desarrolladas, revisando el código y charlando sobre el flujo de datos en la aplicación.

*CRUD: Crear, Leer, Actualizar y Borrar [+]