**:cohete:  Final Project  :cohete:**  
**@aquí** En este proyecto, vais a construir una aplicación con un [CRUD](https://www.crehana.com/blog/transformacion-digital/que-es-crud/)tipo [ToDo List](https://www.google.com/search?q=ToDo+List&hl=es&sxsrf=ALiCzsYMshSDn3IStBoShTo7lvN5Mc6pgw:1666014190090&source=lnms&tbm=isch&sa=X&ved=2ahUKEwiAyPa_suf6AhVX_4UKHYiOCQ4Q_AUoAXoECAEQAw&biw=1920&bih=969&dpr=1) con **Vue.js**.  
La app estará vinculada a una base de datos, donde se almacenarán los datos de los usuarios y sus respectivas tareas. La base de datos que utilizaremos para este proyecto es **Supabase**. También usaremos **Vite** como herramienta de construcción para compilar nuestro código y proporcionar un servidor de desarrollo mientras trabajamos.  
El **objetivo principal** de este proyecto es asegurar **que tu aplicación funcione perfectamente**, pero si tienes tiempo, puedes darle el estilo que quieras para obtener puntos extra.**Requisitos del proyecto**  
:uno:  Debe haber una**pantalla de registro/inscripción**(Sign up / Log in) que se mostrará cuando el usuario no haya iniciado sesión.

* El formulario de registro (Sign up) solicite al usuario una dirección de correo electrónico, una contraseña y la confirmación de la misma.
* El formulario de inicio de sesión (Log in) solicite un correo electrónico y una contraseña al usuario.

:dos:  Tras iniciar la sesión, la aplicación debe **obtener y mostrar todas las tareas** de la lista de tareas del usuario conectado**almacenada en Supabase.**  
:tres:  Vuestra aplicación debe permitir:

* **Añadir**tareas nuevas  **:signo_de_suma_grueso:**
* **Editar**las tareas existentes  **:nota:**
* **Marcar**las tareas como completas/incompletas **:marca_de_verificación_blanca:**
* **Eliminar**tareas existentes (tanto completas como incompletas)  **:la_basura_en_su_lugar:**

:cuatro:  Es **IMPRESCINDIBLE**que hagáis al menos un**COMMIT/PUSH diario** a vuestro repositorio en GiHub. No obstante, es más que recomendable que hagáis commits con cierta frecuencia, es una buena práctica que vale la pena que incorporéis cuanto antes a vuestro proceso de trabajo. Y hablando de [buenas prácticas…](https://midu.dev/buenas-practicas-escribir-commits-git/)  
:cinco:  Vuestra aplicación debe estar desplegada en línea usando**GitHub Pages**.**And remember… *KISS***  
(Keep It Simple, Stupid)

Por si queréis inspiración. **INSPIRACIÓN**!  
Las webs tienen Copy Right '¬¬  
(avisar si hay bug o errores que los solucionaremos)  
[Remember to Git - Laura](https://remember-to-git.vercel.app/auth/login)  
[Procrastinan't - Adrià](https://final-project-ih.vercel.app/)

PENDIENTE HACER!

PREGUNTAR!!!

* Mantener posición del ratón después de actualizar??
* Que se actualice más natural, usar el

import { useTaskStore } from "../store";

const tasks = useTaskStore();

const taskArray = await getTasks();

tasks.tasks = taskArray;

en el eliminar y checkear tasks para que se actualicen como cuando hago addtask.

PROBADO Y NO FUNCIONA BIEN

GENERAL:

* SE PUEDE HACER QUE LA VIEW EN LA QUE ESTOY, TENGA UN BACKGROUND DIFERENTE? O ESTÉ SUBRAYADO O LO QUE SEA, PERO DIFERENTE AL RESTO



QUE QUEDE ASI



* Se puede quitar que al hacer click no salga el cuadrado alrededor?