

## Тестове завдання: “Task Flow Builder”

Вам необхідно створити простий інтерфейс для побудови бізнес-процесів із задачами у вигляді блоків, що з'єднуються між собою (flow chart style).

## Вимоги:

- Візуалізація задач:**
  - Кожна задача — це блок (node) з назвою
  - Можна перетягувати блоки (drag & drop) в межах області.
- Додавання задач:**
  - Кнопка "Add Task" створює нову ноду на полотні (React Flow)
  - Назву задачі можна редагувати в input полі прямо в ноді або в боковій панелі.
- З'єднання задач:**
  - Користувач може з'єднати задачі між собою стрілками (edges).
- Редагування задачі:**
  - При кліку на блок відкривається бокова панель з інпутом для редагування назви.
- Збереження стану:**
  - Використати **Redux Toolkit** для збереження списку задач та зв'язків між ними.
- (Опційно)**
  - Збереження у localStorage, щоб після перезавантаження все зберігалось.

## Технічні вимоги:

- Структурувати компоненти логічно (наприклад: **components/**, **store/**, **types/**).
- Використовувати **useAppSelector**, **useAppDispatch** для роботи з Redux.
- Забезпечити типізацію за допомогою TypeScript.
- Оформити проект мінімально, але охайно.

## Стек технологій:

- React
- Redux Toolkit
- React Flow
- TypeScript
- (Опційно: styled-components / Tailwind / SCSS)

## Що оцінюється:

- Розуміння React/Redux архітектури.
- Вміння працювати з бібліотекою React Flow.
- Читабельність і структура коду.
- TypeScript-типізація.
- Робоча логіка з додаванням, редагуванням та з'єднанням нод.

## Що здати:

- Посилання на GitHub репозиторій.
- Короткий опис, як запустити (**README.md**)
- (Опційно) Демо через Vercel/Netlify.



Володіння чатом gpt вітається, але для виконання цього завдання не вітається 😊  
На виконання завдання Ви маєте 3 дні! Успіхів!