Feladat ütemezés

# Adatbázis

## **users** (felhasználók) tábla

* + - id (PRIMARY KEY, AUTO\_INCREMENT)
    - name (VARCHAR(100))
    - email (VARCHAR(100))
    - created\_at (DATETIME)

## **events** (események) tábla

* + - id (PRIMARY KEY, AUTO\_INCREMENT)
    - title (VARCHAR(200))
    - description (TEXT)
    - date (DATETIME)
    - user\_id (FOREIGN KEY → users.id)

# Backend feladat

Készítsd el az alábbi REST API-kat Express keretrendszerrel, amely az alábbi CRUD műveleteket biztosítja

## Felhasználók

* + - GET /users: Listázza az összes felhasználót.
    - GET /users/:id: Egy felhasználó adatainak lekérdezése.
    - POST /users: Új felhasználó hozzáadása.
    - PUT /users/:id: Egy felhasználó adatainak frissítése.
    - DELETE /users/:id: Felhasználó törlése.

## Események

* + - GET /events: Listázza az összes eseményt.
    - GET /events/:id: Egy esemény adatainak lekérdezése.
    - POST /events: Új esemény hozzáadása.
    - PUT /events/:id: Egy esemény adatainak frissítése.
    - DELETE /events/:id: Esemény törlése.

# React Frontend Felépítése

Egy React alkalmazás biztosítja a felhasználói felületet, ahol az alábbi funkciók érhetők el

## Felhasználók kezelése

* + - Lista megjelenítése a felhasználókról.
    - Új felhasználó felvitele egy űrlap kitöltésével.
    - Felhasználó adatainak szerkesztése.
    - Felhasználó törlése.

## Események kezelése

* + - Lista megjelenítése az eseményekről.
    - Új esemény felvitele.
    - Esemény adatainak szerkesztése.
    - Esemény törlése.

# Backend-Frontend Kapcsolat

A React alkalmazás Axios-t használjon az API hívások végrehajtására. A felhasználóbarát működés érdekében biztosítani kell:

* + - Hibakezelést az API hívásokra.
    - Betöltési állapotokat (“loading” indikátor).
    - Értesítő üzeneteket (pl. sikeres mentés vagy hibaüzem).

# Telepítés és Tesztelés

* + **Adatbázis:** Az adatbázis helyben fut.
  + **Backend:** Az Express alkalmazás futtatása egy Node.js környezetben.
  + **Frontend:** A React alkalmazást hosztold lokálisan vagy egy fejlesztői szerveren.