

1. Backend (Laravel) – 20 pont

Adatbázis (4 pont)

Hozz létre egy `tasks` táblát az alábbi oszlopokkal:

- `id` (auto-increment, primary key)
- `title` (VARCHAR, max 255 karakter) - A feladat megnevezése
- `description` (TEXT, opcionális) - A feladat leírása
- `status` (ENUM: pending, in_progress, completed, archived) - A feladat állapota
- `priority` (ENUM: low, medium, high) - A feladat prioritása
- `due_date` (DATETIME, opcionális) - Határidő
- `created_at`, `updated_at` (timestamp)

Model és kapcsolatok (2 pont)

- Készíts egy `Task` modellt az adatbázis-kezeléshez!
- Hozz létre egy `User` modellt, és állíts be egy kapcsolatot a `tasks` táblával (minden feladat egy felhasználóhoz tartozik).

Controller és API végpontok (10 pont)

Készíts egy `TaskController`-t REST API endpointokkal:

- `GET /api/tasks` – Listázza az összes feladatot.
- `GET /api/tasks/{id}` – Egy adott feladat lekérése.
- `POST /api/tasks` – Új feladat hozzáadása.
- `PUT /api/tasks/{id}` – Feladat módosítása.
- `PATCH /api/tasks/{id}/status` – Feladat állapotának módosítása.
- `DELETE /api/tasks/{id}` – Feladat törlése.

- Az adatok validálását is végezd el (pl. kötelező mezők, ENUM értékek ellenőrzése).
- Az API autentikációt használjon (pl. Laravel Sanctum vagy Passport).

Seeder és Migration (4 pont)

- Hozz létre migrációt az adatbázis-struktúra generálására!
 - Készíts egy seeder fájlt, amely létrehoz 5 minta felhasználót és mindegyikhez legalább 3 feladatot!
-

2. Frontend (Angular) – 30 pont

Alkalmazás felépítése (3 pont)

- Hozz létre egy új Angular projektet `taskManager` néven!
- Konfiguráld az `HttpClientModule`-t az API hívásokhoz!

TaskService API-val (10 pont)

Készíts egy `TaskService`-t az API műveletekhez:

- `getTasks()` – Összes feladat lekérése
- `getTaskById(id: number)` – Egy adott feladat lekérése
- `addTask(task: Task)` – Új feladat létrehozása
- `updateTask(task: Task)` – Feladat módosítása
- `updateTaskStatus(id: number, status: string)` – Feladat állapotának módosítása
- `deleteTask(id: number)` – Feladat törlése

Task Komponens (10 pont)

Készíts egy komponenst (`TaskComponent`), amely:

- Listázza a feladatokat

- Lehetővé teszi új feladat hozzáadását
- Lehetővé teszi a meglévő feladatok módosítását és törlését
- Szűrési lehetőséget biztosít állapot (**status**) és prioritás (**priority**) szerint

Formok és validáció (5 pont)

- Az űrlap validálja a mezőket:
 - A **title** kötelező és max 255 karakter lehet.
 - A **status** csak a megadott ENUM értékeket fogadja el.
 - A **priority** csak a megadott ENUM értékeket fogadja el.
 - A **due_date** nem lehet múltbeli dátum.

Extra funkció (2 pont)

- Készíts egy külön **TaskFilterComponent**-et, amely segít a feladatok szűrésében állapot és prioritás szerint!