



L6: Authentication & Authorization

- Az alábbi kiegészítéseket véghez víve vezessünk be **hitelesítést** és **engedélyezést** a korábbi laborokon készített projektbe.
- Továbbra is feltételek a következők:
 - Ne legyenek ESLint/Prettier/HTML5 warningok/errorok a kódban s legyen zöld az utolsó pipeline kimenete.
 - Minden node.js függőség legyen listázva a `package.json` állományban.
 - Hagyjunk kommentben vagy külön állományban (lehetőleg a `README.md`-ben) leírást arról, hogy hogyan lehet a rendszert működésre bírni (pl. milyen adatbázist kell elindítani, felhasználót készíteni rajta, esetleg táblákat létrehozni, stb.).
 - A kód legyen helyesen **modularizálva** – minden modul tartsa be a *single responsibility principle*-t.
- Az első és második rész általánosan érvényes mindenkre, s mindenki alkalmazza őket azokra az entitásokra/funkcionalitásokra, melyek már léteznek a korábbi laborokról. Ha valahol az általános leírás nem tisztán alkalmazható a meglévő projektre, kérdezzétek a laboránsokat tisztázásért. A harmadik rész személyre szabott a diákazonosító szerint.

Első rész: Be- és kijelentkezési felület

Be- és kijelentkezési felület

- Adaptáljuk a kliensoldalunk **minden oldalát** a következőképpen:
 - Tisztán látszodjon, hogy egy felhasználó *be van-e jelentkezve*, s ha igen, ki ő (pl. felhasználónév, teljes név vagy e-mail cím).
 - Ha nincs bejelentkezve, egy „Login” gomb/link irányítson át (vagy jelenítsen meg a helyi oldalon) egy bejelentkezési felületre, melyen keresztül megfelelő információ megadásával jelentkezessen be.
 - Ha be van jelentkezve, egy „Logout” gomb/link hajtsa végre a kijelentkezést.
- A felületek kódja ne ismétlőjön emiatt – használjunk pl. `partial`okat.

Második rész: Szerveroldali hitelesítés

Szerveroldali hitelesítés

- Ellenőrizzük az adatbázisunk *felhasználó* tábláját (hozzuk létre ha még nem létezik). Győződjünk meg, hogy ez tárol egy *hash*-elt jelszót.
- Implementáljuk a be- és kijelentkezési folyamatot a szerverünkön. Az információkat tárolhatjuk:
 - (csak szerveroldalról elérhető) *JWT* sütiben
 - szerveroldali munkamenetben (*session*)
- Készítsünk egy vagy több *middleware*-t, mely ellenőrzi egy felhasználó megfelelő bejelentkezetségi állapotát. Egy hibás bejelentkezetségi állapotot kezeljünk megfelelő hibaüzenetekkel és/vagy átirányítással.

Harmadik rész: Entitások tulajdonosa

Szerveroldali hitelesítés és engedélyezés

- Az alábbiak alapján valamely korábban bevezetett entitásnak legyen tulajdonosa a *bejelentkezett felhasználó*–létrehozáskor ez automatikusan legyen beállítva.
- Későbbi írási műveletek (pl. fénykép-hozzáadás, törlés) csak a tulajdonosnak legyenek megengedve, másnál ne is jelenjenek meg az *afferens* gombok/input mezők.
- Feladatonkénti konkrétumok alább.

6.1. Online napló

- A tantárgyat létrehozó személy automatikusan ennek tulajdonosává válik.
- Egy tantárgyhoz csak a tulajdonos tölthet fel vagy törölhet feladatot, más bejegyzéseknél nem jelenik meg a feltöltési form, illetve a „Törlés” gomb.

6.2. Használt autó kereskedés

- A hirdetést létrehozó személy automatikusan ennek tulajdonosává válik.
- Egy hirdetéshez csak a tulajdonos tölthet fel vagy törölhet képet, más bejegyzéseknél nem jelenik meg a feltöltési form, illetve a „Törlés” gomb.

6.3. Online filmgyűjtemény

- A visszajelzés automatikusan a bejelentkezett felhasználó nevében jön létre, legördülő listából való kiválasztás helyett.
- Csak a tulajdonos törölheti a visszajelzést, más bejegyzéseknél nem jelenik meg a „Törlés” gomb.

6.4. Sportközpont pályafoglaló

- A foglalás automatikusan a bejelentkezett felhasználó nevében jön létre, legördülő listából való kiválasztás helyett.
- Csak a foglaló személy törölheti a foglalást, más bejegyzéseknél nem jelenik meg a „Törlés” gomb.

6.5. Órarendkészítő

- A tantárgyat létrehozó személy automatikusan ennek tulajdonosává válik.
- Egy tantárgyhoz csak a tulajdonos tölthet fel vagy törölhet erőforrás-állományt, más bejegyzéseknél nem jelenik meg a feltöltési form, illetve a „Törlés” gomb.

6.6. Online könyvtár

- A kölcsönzési és visszaszolgáltatási folyamat a bejelentkezett felhasználó nevében zajlik, legördülő listából való kiválasztás helyett.
- A bejelentkezett felhasználó külön listázhatja a saját aktív foglalásait.

6.7. Rendezvényszervező

- A csatlakozási és kilépési folyamat a bejelentkezett felhasználó nevében zajlik, legördülő listából való kiválasztás helyett.
- A bejelentkezett felhasználó külön listázhatja azokat a rendezvényeket, amelyek esetén ő is szervező.

6.8. Vonattársaság

- A foglalás automatikusan a bejelentkezett felhasználó nevében jön létre, legördülő listából való kiválasztás helyett.
- Csak a foglaló személy törölheti a foglalást, más bejegyzéseknél nem jelenik meg a „Törlés” gomb.

6.9. Lakáshirdetések

- A hirdetést létrehozó személy automatikusan ennek tulajdonosává válik.
- Egy hirdetéshez csak a tulajdonos tölthet fel vagy törölhet képet, más bejegyzéseknél nem jelenik meg a feltöltési form, illetve a „Törlés” gomb.

6.10. Asztalfoglaló vendéglőknek

- A foglalás automatikusan a bejelentkezett felhasználó nevében jön létre, legördülő listából való kiválasztás helyett.
- Csak a foglaló személy törölheti a foglalást, más bejegyzéseknél nem jelenik meg a „Törlés” gomb.