

A	Név:	Neptun kód:
Gép sorszám:	Értékelő oktató:	Jegy:

Feladat: Hallgatói tananyagmegosztó portál

Készítsünk egy olyan weblapot ASP.NET Core MVC keretrendszer használatával, amely elősegíti az egyetemi tananyagok, jegyzetek hallgatók közötti megosztását. Az adatokat a rendszer adatbázisban tárolja, amelynek szerkezete *code-first* módon definiálandó. Az adatbázis az alábbi adatokat tárolja:

- tantárgyak (név)
- tananyagok (cím, tantárgy, feltöltött állomány, időbélyeg, letöltések száma)
- értékelések (tananyagok, érték)

Részfeladatok:

1. **(2 pont)** A weblap a feltöltési idő szerint csökkenő sorrendben listázza a feltöltött tananyagokat, lehetővé téve letöltésüket. Legyen lehetőség új tananyag feltöltésére a cím megadásával, a tantárgy kiválasztásával (legördülő menüből), valamint egy fájl feltöltésével.

A tantárgyak kezelése (létrehozása, törlése) nem feladat, néhány tantárgy az adatbázissal együtt inicializálható.

2. **(1 pont)** A weblap lapozható formában korlátozza az egy oldalon listázott tananyagok számát. Legyen lehetőség cím- és tantárgynév-részletre keresve szűrni.
3. **(1 pont)** A weblapon a látogatók egy skálán (1-5) értékelhessék a feltöltött tananyagok minőségét, hasznosságát. A weblap a kilistázott anyagok mellett jelenítse meg az értékelések számtani közepét.
4. **(1 pont)** A rendszer számolja, hogy az egyes tananyagokat hányszor töltötték már le, és ezt az értéket jelenítse is meg a tananyagok kilistázásakor.

A megoldást az elfogadást követően **ZIP** formátumban fel kell tölteni a <https://assignment.elte.hu/> beadandókezelő rendszerbe.

Jó munkát!

B	Név:	Neptun kód:
Gép sorszám:	Értékelő oktató:	Jegy:

Feladat: Vállalati hibabejelentő portál

Készítsünk egy olyan weblapot ASP.NET Core MVC keretrendszer használatával, amely elősegíti egy cégnél előforduló hibajelenségek bejelentését. Az adatokat a rendszer adatbázisban tárolja, amelynek szerkezete *code-first* módon definiálandó. Az adatbázis az alábbi adatokat tárolja:

- hibajelenségek (cím, leírás, kép, időbélyeg, kérések száma)
- felhasználók (felhasználónév, email cím, jelszó hash)
- kérések (felhasználó, hibajelenség)

Részfeladatok:

- 1. (2 pont)** A weblap listázza ki az elmúlt 30 nap bejelentett hibajelenségeit. (Amelyik probléma nem oldja meg magát 30 nap alatt, azzal nem érdemes foglalkozni.) Jelenjen meg az egyes bejelentett hibák címe, leírása, bejelentés időpontja, valamint a hozzájuk opcionálisan feltöltött kép is. A bejelentések legyenek az időbélyeg szerint csökkenően rendezve. Legyen lehetőség új hiba beküldésére a cím és a leírás megadásával, valamint opcionálisan egy kép feltöltésével. Kerüljön ellenőrzésre, hogy a feltöltött állomány valóban képfájl-e.
- 2. (1 pont)** A weblap látogatói „+1”-gyel szavazhassanak az adott hibajelenségekre, jelezve, hogy javításukat ők is sürgősnek tartják. A rendszer tartsa nyilván, hogy az egyes hibák javítását hányan kérik (kezdeti értéke bejelentéskor legyen 1), és az elmúlt 30 nap hibajelenségeit „népszerűségük” szerint csökkenő rendben listázza.
- 3. (1 pont)** A vállalat alkalmazottai azonosító és jelszó páros megadásával jelentkezhetnek be, a jelszó biztonságos (hasított) tárolásáról gondoskodni kell. Új hibát beküldeni, valamint meglévő hibára szavazni csak bejelentkezett felhasználók tudjanak. A felhasználók regisztrálása nem feladat, néhány minta felhasználó az adatbázissal együtt inicializálható.
- 4. (1 pont)** A rendszer tartsa nyilván, hogy az egyes hibák javítását kik kérték (kik szavaztak rá), és garantálja, hogy 1 felhasználó 1 hibára csak 1-szer szavazhasson. (A hiba bejelentése egyben számítson szavazásnak továbbra is)

A megoldást az elfogadást követően **ZIP** formátumban fel kell tölteni a <https://assignment.elte.hu/> beadandókezelő rendszerbe.

Jó munkát!

C	Név:	Neptun kód:
Gép sorszám:	Értékelő oktató:	Jegy:

Feladat: Egyetemi katalógus rendszer

Készítsünk egy egyetemi katalógus webalkalmazást ASP.NET Core MVC keretrendszer használatával. Az adatokat a rendszer adatbázisban tárolja, amelynek szerkezete *code-first* módon definiálandó. Az adatbázis az alábbi adatokat tárolja:

- felhasználók (egyetemi azonosító, email cím, jelszó hash, oktató-e)
- meghirdetések (tantárgy neve, oktató, ellenőrző kód, kezdő időpont, időtartam)
- bejelentkezések (felhasználó, meghirdetés)

Részfeladatok:

- 1. (2 pont)** Az egyetem oktatói azonosító és jelszó páros megadásával jelentkezhetnek be, a jelszó biztonságos (hasított) tárolásáról gondoskodni kell. Az oktatóként bejelentkező felhasználók készíthessenek új meghirdetést a jelenlét ellenőrzésére egy tetszőleges tantárgynév, valamint az időtartam (5, 10 vagy 20 perc) megadásával. A kezdő időpont az aktuális időbélyeg legyen.
A felhasználók regisztrálása nem feladat, néhány minta hallgató és oktató az adatbázissal együtt inicializálható.
- 2. (1 pont)** Új meghirdetés készítésekor a rendszer készítsen egy 6 karakter hosszú, véletlen generált ellenőrző kódot (*token*). Az oktatói felületen legyen lehetőség az adott oktató által utoljára készített meghirdetés tantárgynevének és ellenőrző kódjának megtekintésére.
- 3. (1 pont)** Az egyetem hallgatói is jelentkezhetnek be azonosító és jelszó páros megadásával. A hallgatói felületen kerüljenek kilistázásra az aktuálisan élő meghirdetések.
- 4. (1 pont)** A hallgatóknak legyen lehetősége az aktív meghirdetésekre jelentkezni (a katalógust „aláírni”), az adott meghirdetéshez tartozó ellenőrző kód megadásával. A rendszer ne engedélyezze a duplikált jelentkezést. (Feltételezhetjük, hogy a hallgatók nem jelentkeznek be egymás nevében.)

A megoldást az elfogadást követően **ZIP** formátumban fel kell tölteni a <https://assignment.elte.hu/> beadandókezelő rendszerbe.

Jó munkát!