

# Loyiha nomi: Online Kurslarni Boshqarish Tizimi

## Texnik topshiriq:

Sizga **Online Kurslarni Boshqarish Tizimi** yaratish kerak. Ushbu tizim foydalanuvchilarga kurslarni ko'rish, ularga yozilish, va kursdagi mashg'ulotlarni bajarish imkoniyatini beradi. Adminlar yangi kurslar qo'shishi, o'zgartirishi va o'chirishi mumkin. Tizim o'quvchilar uchun ro'yxatdan o'tish va tizimga kirish imkoniyatini taqdim etadi.

## Talablar:

1. **Backend:** Loyiha NestJS'da yozilishi kerak.
2. **Ma'lumotlar ombori:** PostgreSQL. Yoki Mongo
3. **Autentifikatsiya:** JWT orqali amalga oshiriladi.
4. **Konteynerizatsiya:** Docker orqali ilovangizni konteynerga joylashtirishingiz kerak.
5. **Versiya nazorati:** GitHub'da alohida branch yaratib ishlashingiz kerak, har bir qism uchun commitlar qo'shib boring.

## Asosiy funksiyalar:

1. **Foydalanuvchi (student) ro'yxatdan o'tishi va tizimga kirishi:**
  - Foydalanuvchilar **name**, **email**, va **password** orqali ro'yxatdan o'tishlari kerak.
  - JWT yordamida autentifikatsiya amalga oshiriladi.
2. **Admin panel:**
  - Admin yangi kurslar qo'shishi, ularni tahrirlashi va o'chirishi mumkin.
  - Har bir kurs quyidagi ma'lumotlarni o'z ichiga oladi: **name**, **description**, **price**, **teacher**, **category**, va **level**.
3. **Kurslarni ko'rish:**
  - Foydalanuvchilar barcha kurslarni ko'rishlari mumkin.
  - Kurslar kategoriyalar bo'yicha filtrlash va qidirish imkoniyatiga ega bo'lishi kerak.
4. **Kursga yozilish:**

- Foydalanuvchi kursga yozilishi mumkin, kursga yozilgandan keyin u kursning barcha mashg'ulotlarini ko'rish imkoniyatiga ega bo'ladi.

**5. Mashg'ulotlar (modules):**

- Har bir kurs bir nechta modullardan iborat bo'lishi mumkin.
- Har bir modulda bir nechta darslar (lessons) bo'lishi kerak.
- Har bir dars video yoki matn ko'rinishida bo'lishi mumkin.

**6. Topshiriqlar (assignments):**

- Har bir modul yakunida foydalanuvchi topshiriq bajarishi kerak.
- Topshiriqlar natijalari avtomatik ravishda baholanadi.
- Topshiriqlarni admin yoki o'qituvchi qo'lda baholaydi.

**7. Baholash va natijalar:**

- Testlar va topshiriqlar bahosi **Results** sahifasida ko'rsatiladi.
- Har bir kurs uchun foydalanuvchi umumiy natijasini ko'rishi mumkin.

**API routlari(routes):**

**1. Foydalanuvchi:**

- **POST /auth/register**: Foydalanuvchi ro'yxatdan o'tadi.
- **POST /auth/login**: Foydalanuvchi tizimga kiradi.
- **POST /auth/logout**: Foydalanuvchi tizimdan chiqadi.

**2. Kurslar:**

- **GET /courses**: Kurslarni ko'rish.
- **POST /courses**: Yangi kurs qo'shish (faqat admin).
- **PUT /courses/:id**: Kursni tahrirlash (faqat admin).
- **DELETE /courses/:id**: Kursni o'chirish (faqat admin).

**3. Modullar va darslar:**

- **GET /courses/:courseId/modules**: Kursdagi barcha modullarni ko'rish.
- **GET /modules/:moduleId/lessons**: Moduldagi darslarni ko'rish.

**4. topshiriqlar:**

- **POST /modules/:moduleId/assignment**: Topshiriqni topshirish.

**5. Natijalar:**

- **GET /results**: Foydalanuvchining umumiy natijalarini ko'rish.

**Texnik talablar:**

1. **NestJS modullari:** Har bir qism uchun (auth, courses, modules, lessons, assignments, results) alohida modul yarating.
  2. **TypeScript:** Kodingizda TypeScript imkoniyatlarini to'liq qo'llang.
  3. **PostgreSQL:** Ma'lumotlar bazasini PostgreSQL orqali boshqaring. Alohida tablitsalar orqali kurslar, modullar, darslar, testlar va foydalanuvchilarni tashkil eting.
  4. **Docker:** Ilovani Docker konteyneriga joylashtiring. **docker-compose** orqali PostgreSQL va NestJS ilovasini birgalikda ishlating.
  5. **Git:** Git'da **main** branch'ga ishlamang, alohida branch yaratib ishlang. O'zgarishlarni har safar **commit** qilib, GitHub'da saqlang.
- 

#### **Baholash mezonlari:**

1. Funksional talablarning bajarilishi: Ilova berilgan talablar asosida to'liq ishlashi kerak - **35 ball**.
2. Kod sifati: TypeScript'ni to'g'ri qo'llanishi, NestJS modullari tuzilmasi, toza va o'qilishi oson bo'lgan kod - **30 ball**.
3. Docker integratsiyasi: Ilovani Docker orqali muvaffaqiyatli ishga tushirish - **20 ball**.
4. Git commit'lari: Kichik-kichik o'zgarishlar qilinib, muntazam commit'lar qo'yilgan bo'lishi kerak - **15 ball**.

#### **Imtihonga topshiriladigan resurslar :**

1. Qilingan projectni to'liq kodi: github link shaklida yoki zip file qilib.
2. Postmant to'liq yozilgan bo'lishligi . Ayniqsa men yuqirida aytib o'tgan hamma routlar bo'lishi kerak . - **15 ball** .
3. Konsoleda ma'lumot qolib ketmasin ortiqcha . Modellarni ulashda juda e'tiborli bo'linglar .
4. Umimiy beriladigan vaqt: **1 hafta**
5. Kimdir deploy qilsa **10 ball** +

*Hammaga omad tilab qolamiz !!!*

*Hurmat bilan Najot ta'lim jamoasi .*

