

Gittigidiyor Spring Bootcamp Homework 04

Aşağıdaki iş kuralı senaryosuna göre bir önceki ödevlerde hazırladığınız Spring Boot projesini aşağıdaki sorulara göre tekrar düzenleyiniz.

ENGLISH	TÜRKÇE
School Management System <ul style="list-style-type: none">A Student who has a name, a birth date, an address, and gender could take zero or more courses.Each course has a course name, a course code, and a credit score.Instructors who have a name, an address and a phone number could instruct at least one or more courses.A course could be instructed by only one instructor.There are 2 types of instructors; “visiting researcher” or “permanent instructor”.A visiting researcher takes an hourly salary, but a permanent instructor takes a fixed salary. <p>Example: Student → Class candidate name → variable candidate zero or more → cardinality candidate</p>	Okul Yönetim Sistemi <ul style="list-style-type: none">Adı, doğum tarihi, adresi ve cinsiyeti olan bir öğrenci hiç ders almaz veya bir yada birden fazla ders alabilir.Her dersin bir ders adı, ders kodu ve kredi puanı vardır.Adı, adresi ve telefon numarası olan eğitmenler en az bir veya daha fazla ders verebilir.Bir kurs yalnızca bir eğitmen tarafından verilebilir.2 tür eğitmen vardır; “Misafir araştırmacı” veya “daimi eğitmen”.Misafir araştırmacı saatlik maaş alır, ancak daimi eğitmen sabit maaş alır. <p>Örnek: Öğrenci → Java sınıfı (model) adayı isim → değişken adayı en az bir veya daha fazla → azlık/çokluk adayı</p>

SORU-1 : Controller katmanında request ve response nesnesi olarak “Data Transfer Object (DTO)” kullanılacaktır.

SORU-2 : DTO ve Entity dönüşümleri (mapping) Mapstruct API kullanılarak gerçekleştirilecektir.

SORU-3 : Aşağıdaki yeni iş kurallarına uygun gerekli controller/service/repository düzenlemeleri yapılacaktır;

- Öğrenci yaşı 18’den küçük, 40’tan büyük olamaz. Aksi halde **StudentAgeNotValidException** hatası fırlatılır.
- Sistemde aynı telefon numarasına sahip 2 ayrı eğitmen bulunamaz. Tanımlanması durumunda **InstructorAlreadyExistException** hatası fırlatılır.
- Sistemde aynı ders koduna sahip 2 ayrı ders olamaz. Tanımlanması durumunda **CourseAlreadyExistException** hatası fırlatılır.
- Bir dersi aynı anda en fazla 20 öğrenci alabilir. Aksi halde **StudentNumberForOneCourseExceededException** hatası fırlatılır.
- Tüm hata durumları, Global Exception Handler sınıfı ile yakalanarak uygun **status code** ve **message** ile oluşturulacak ve response olarak dönecektir.

SORU-4 : Soru-3'te belirtilen hatalar, fırlatılma tarihi ve hata mesaj bilgisi ile birlikte database içerisinde loglanacaktır (ayrı bir tablo oluşturunuz). Ayrı bir endpoint hazırlanacak ve loglanmış geçmiş hata bilgileri, hata tipi ve/veya tarihe göre sorgulanabilecektir.

SORU-5 : Tüm entitylerin create ve modify zamanlarını database üzerinde tutacak şekilde gerekli düzenleme yapılacaktır.

SORU-6 : Tüm sistemin fonksiyoneliyetini ve iş kurallarını açıklayacak şekilde bir **README.md** dosyası projeye eklenecektir (İstenirse ekran resimleri eklenebilir). Kullanılan metodlara **javadoc** standartları çerçevesinde commentler eklenecektir.

ARAŞTIRMA KONUSU : JVM nedir ve nasıl çalışır? Mimarisi nasıldır? Java 8 ile birlikte JVM mimarisinde ne gibi değişiklikler olmuştur?

MEET-UP TAVSİYESİ : Trendyol ile Spring Native & GraalVM'i tanıyoruz ([Link](#)) – [02 Eylül 2021 Perşembe 18:30 - 19:30]