ابتدا مدل های خام ساخته شد. [create raw models](https://github.com/negarnegma/studentcoursemastermanage/commit/5a4114915b800a4ef3127606a7b06935435d0882)  
پکیج model

BaseEntity برای جلوگیری از تکرار کد است. ویژگی هایی که همه ی مدل ها دارند.  
StudentCourse برای ارتباط بین دانش آموز و درس. به علت داشتن فیلدی اضافه به نام نمره، این کلاس تعریف شد وگرنه به هایبرنیت سپرده می شد.   
  
پکیج modelVO

VO = Value object

@JsonProperty("last-name")

این یعنی وقتی آبجک رو تبدیل به جیسون کردیم و فرستادیم یا جیسون گرفتیم و خواستیم تبدیل کنیم؛ این فیلد رو با last-name بشناسیم. حالا اگه اسم قبلیش(اسم خود فیلد) رو بدیم قبول هم نمیکنه!!  
-------------------------------

وب سرویس برای تعامل با برنامه ساخته شد. [added webservice support](https://github.com/negarnegma/studentcoursemastermanage/commit/59457d94a50de0a129bbc2720879854f307ff27f)

(با استفاده از فریمورک jersey)

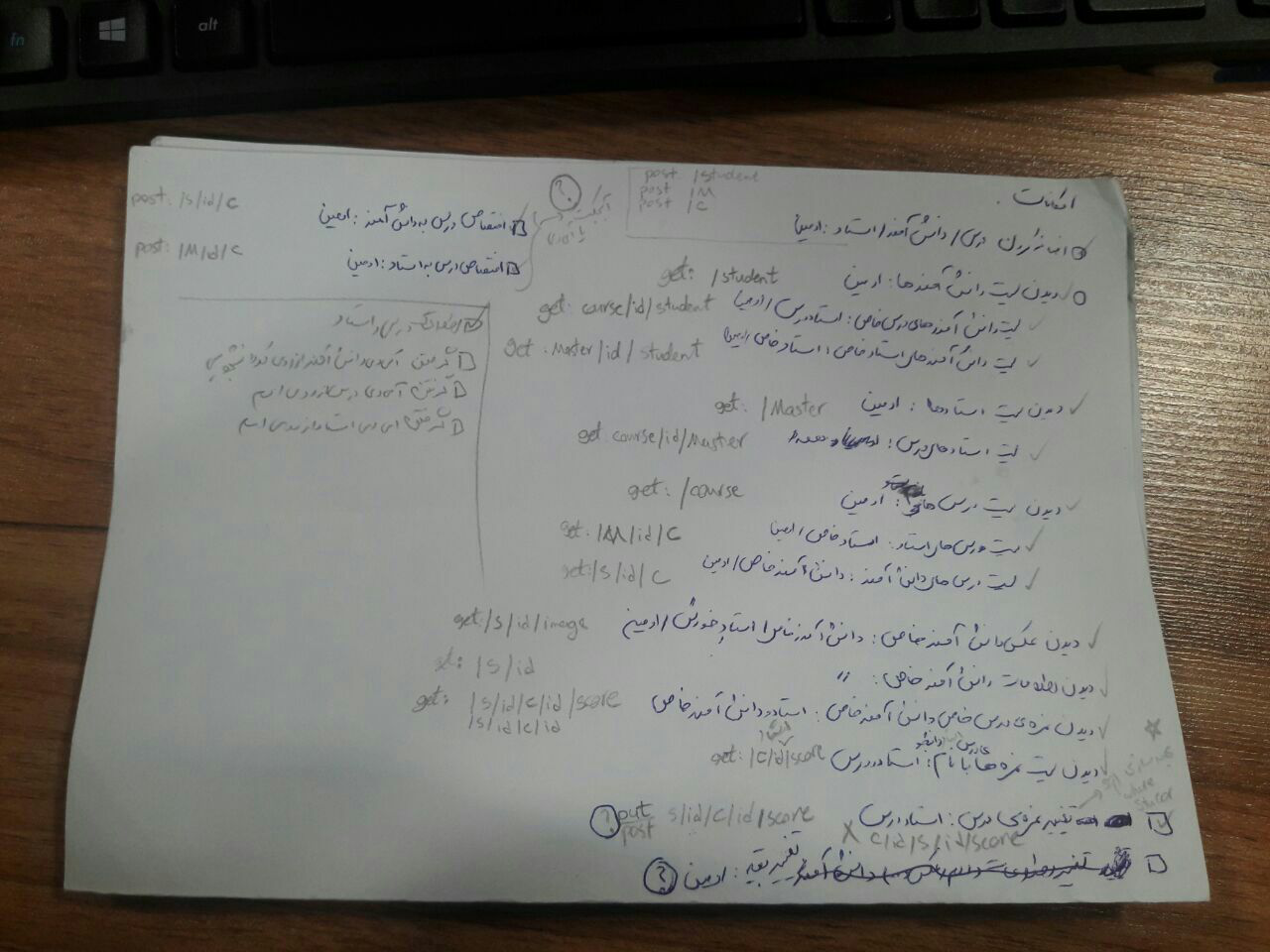
پکیج controller آپدیت شد.

Dependency ها به pom.xml اضافه شد.

تا اینجا اگر برنامه را با postman یا مرورگر امتحان کنید کار میکند. اما چون هنوز دیتابیس اضافه نشده؛ و همچنین CRUD ها پیاده سازی نشده اند؛ نتایج به درد بخور نیستند. و چیزی هم سیو نمیشود.

به نظرم خوب اومد که FindApi رو هم اضافه کنم تا آی دی ها رو راحت به دست بیاریم.

قبل از پیاده سازی سرویس ها، روی کاغذ نوشتم امکاناتی که لازم دارم رو. بعد با مداد مسیر پیشنهادی برای اون عملیات رو نوشتم (url) بعد هم بر اساس آدرس و مفهوم دسته بندی کردم و api ها رو پیاده سازی کردم.



--------------------------

BeanUtils راهی راحت برای کپی پراپرتی هاست. HibrenateUtils سینگلتون و لیزی است.

تکه کد های زیبا:

استفاده از stream برای حذف داپلیکیت ها:

List<Student> students2 = students.stream()  
 .distinct()  
 .collect(Collectors.*toList*());

استفاده از methoderefrence

List<Long> mids = course.getMasterList().stream().map(BaseEntity::getId).collect(Collectors.*toList*());

استفاده از where های بهینه: ابتدا فیلتر کم تر جدا می شود:

reateQuery("from StudentCourse where student=:student AND course:=course ");

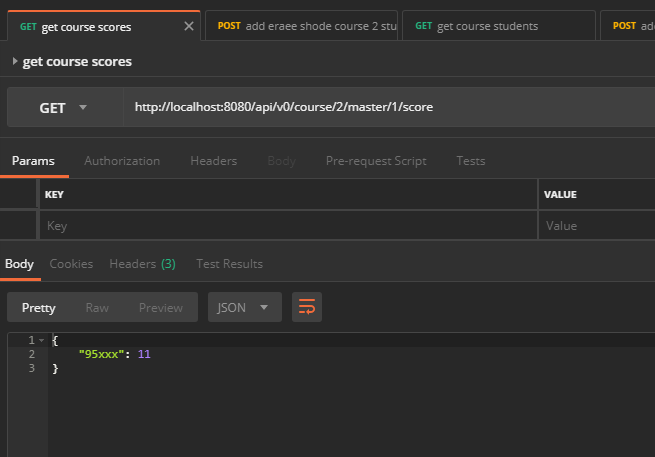
استفاده از اکسپشن های شخصی سازی شده:

if (! course.getMasterList().contains(master))

throw new Exceptions.notMasterInList("this courrse is not eraee n by this master!");

امکانات جالب برنامه:

مشاهده لیست شخصی سازی شده ی نمرات برای درس استاد ( با استفاده از HashMap و نه ایجاد کلاس جدید)



پیدا کردن آی دی ها بر اساس نام و کد دانشجویی (در صورت فراموشی آی دی) (درس/استاد/دانشجو)

-----------------------------

در انتها ~~امنیت~~ و داکیومنت (ابتدا با postman و سپس تبدیل آن به swagger) اضافه شد.