### 第十三周周报

预定目标：在之前了解到了部分前后端分离的思想，后台写接口传数据，而前台采用三大主流框架去操纵数据即可。由于我主修课程为.Net,但是我之前并不太会去写后台的接口，仅仅只能做到数据/界面都是混搭在一起也就是SOAP（单页面应用），在一次偶然的时候了解到了REST的设计思想，从而接触到了Web Api，认为这是我近期需要去学习的方向。

我在b站找了一些资源，筛选后，选择了Dave的学习视频：ASP.NET Core 2.1 + Ng 6 实战，这是一个前后台分离的博客系统，前台搭建采用Angular，后台采用.Net Core WebApi，身份验证采用IdentityServer4，这个项目设计了我很多没有接触到而必须要学习的知识，所以我计划本周跟着这个视频进行学习，下面说一下，本周学习的过程和每天的学习总结以及学习过程中产生的问题。

第一天，刚开始进行学习，就接触到了一个全新的概念EF实体，视频中先通过写Model实体代码，引入EF库进行实体间的约束，然后反向建立数据库，并进行数据迁移。第一天的主要内容是这样的，但是我并没有成功的完成，EF的约束我并没接触太多，而反向生成数据库更是我没有接触的。我上网查了资料，了解到，有三种设计思想，CodeFrist,ModelFirst和DatabaseFirst，而我用的便是CodeFirst（代码先行）。而且在引入EF库的时候，有时候并不能通过内置Nugest直接导入，很是麻烦，这个问题一直困扰这我，直到现在也不知道为什么。所以为了解决EF的配置和使用，我花了一天的时间去观看视频和查找资料，最后成功解决，第一天也就是主要解决了EF的使用。

第二天，由于解决了数据库的创建，学习过程便是很顺利了，文件配置，中间件注册，服务注册都成功完成，并没有什么问题，也让我理解了特性，服务，依赖注入等的使用。下午设计了基本的CRUD接口，经postman测试无异常。期间有个地方不太理解，就是类接口的设计，这一点它们间的联动我并不是太清晰。仅仅是跟着视频敲了下来。还有一个前后台访问的思想，浏览器并不应该直接去调用后台的方法接收数据，为了保护数据，也为了整个框架的稳定性，降低耦合度，应该在浏览器和后台系统之间建立一层实体映射模型，浏览器访问到的实体字段并不是数据库中的字段，而是经过映射后的效果。

第三天，上午跟着视频进行了翻页排序过滤等方法的实现，期间有很多问题，自己并不理解他们的使用要求，时机，意义。于是在进行过滤的时候，不知道是因为类库版本问题，还是环境配置问题，我调用的类并不能使用，一直报错。我尝试去查资料解决，但是并没有用。我也重新比对视频，结果并没有新的发现，看下边的评论，期间也有人遇到这个问题，我也私信了他，然而没有解决办法。这个问题修改到了下午，一直没有成功解决。晚上的时候想着多学些新知识，把这个功能跳过去，进行下一步;项目前台的搭建，然而更是让我头疼，Angular框架我并没有接触，于是踩了很多坑，写的很慢。晚上睡觉的时候觉得这个并不是自己目前能解决的问题，于是我选择了放弃这个项目的学习。晚上觉得直接学习.net core web api有点事倍功半，所以我找了一个MVC的博客项目进行再次学习。

第四天，再次从头开始学习Asp.Net Mvc的学习，之前直接接触web api，并没有用框架去自动搭建，全部都是用手动搭建的，通过MVC的学习，理解并明白了之前的每一步用法。今天还是写后台基本框架，由于有了一点基础，所以写起来很是顺利，并没有什么不理解的地方。直到晚上碰到了一个添加错误，一直添加不到数据库，提示该字段无效。由于快要回寝室了，并没有仔细去排查。留到了第二天。

第五天，针对昨天的问题“该字段无效”，之前也遇到过这个错误，也就是数据库中和代码里的字段不一致导致的，原本以为很简单。我查看了数据库字段和实体间的字段，没有错误，认为不是这里的问题；我按照常规操作，进行断点检查，数据都传输有效，代码也都覆盖了，并没有找出来错误。然后，我又回放视频进行比对，也没有发现错误。然后我下载了视频源码，进行比对，发现有个实体类型字段不一致，然后修改后感觉应该可以了，结果还是不对。此刻已经下午了，我已经快崩溃了，这个问题。我再次回到起点，仔细想了想问题，坚定该字段无效，就是数据库字段和代码实体字段不一致导致的。于是我又从数据库着手，开始进行排除，抱着试一试的心态，修改了数据库中的字段，结果一下就成功了。因为我才用的是代码先行的操作，所以我一直认为数据库字段是没有问题，如果有问题，也是代码实体的字段出错。反正这个简单的错误导致我花费了将近一天的时间，并没有了解到什么新知识。不过在重复代码的过程中，加深了自己的印象。

周六，周日的话这周没有学习相关知识，由于我弟弟生日，所以赶回了家。这周将会进行相关学习，并将这个MVC项目完善。