

2021年7.10-7.23的实习总结

笔记本： 暑期实习

创建时间： 2021/7/24 17:20

更新时间： 2021/7/24 17:25

作者： 134exetj717

URL: <http://wiki.suncaper.net/pages/viewpage.action?pageId=41877548>

实习总是很短暂，但是在这珍贵的时间里，我学到了很多东西。

首先是UML图的绘制。由于开发经验不足，考虑问题没有从实际情况出发。其实时序图、数据流图、思维导图等等这些，应该只是给我们提供基本思路以及项目运行逻辑的工具，应当简短易明，而不是把项目的整体流程每个细枝末节都写出来，这样的话只会让我们的图显得非常复杂，导致实用性不高；

其次是数据库的数据模型设计。学习书本的理论总归和实践不太一样，很多时候根据所学内容的方法来设计表，最后往往发现这样的表是不符合实际情况，在调用的过程中，我们往往需要重新设计一个封装类或者接口，然后实现对应数据提供的功能。因此，通过本次实习我发现，设计数据库的时候应当和对应界面联系起来，界面需要什么数据，那么我们就应该给表格尽量提供这些数据。

再者是统一规范的问题。第一，合作开发项目的过程中，前端人员应该尽量使用前端语言来实现功能，比如xaml，后端人员则应该在cs文件中展开类和函数的封装，提供相应接口，数据库管理人员就应当使用数据库连接语言实现数据提供，由此，开发的文档中格式会非常清晰，非常便于管理以及重新编写或更改；第二，开发过程中，应当设定相关文档，实现三层架构，表现层（UI），业务逻辑层（BLL），数据访问层（DAL）以及实体类库（Model），UI给前端人员使用，BLL给后端提供功能接口的人员使用，DAL给后端实现数据库连接以及增删改的人员使用，Model则提供给全体成员调用；界面风格样式的统一以及布局的统一，界面风格统一化才会看起来赏心悦目不至于混乱，布局的选择统一化有利于修改以及提高设计效率。

然后是git码云的使用。第一，我们需要设定.gitignore文件，在上传之前先确保一些本地无用的文件不会上传，否则会导致每次拉取都需要覆盖非常麻烦；第二，我们应该设定好分支，比如三层架构里面的内容设为三个分支，资源设为资源分支，然后在上传或者拉取的过程中就只需要选择相应的分支，可以大大提高我们的容错率以及效率，通俗地来讲，就是我们拉取不容易犯错，不容易前端修改导致后端修改出错，以及后端修改只需要拉取后端分支，那么效率就高。

最后是，分工问题。项目准备工作阶段，我们就应当设计好原型界面及其需要实现的功能，然后前后端人员就能同时开始有目的的工作，开发过程中，若前端人员产生了新的接口功能需求，那么就只需要分配给后端人员即可。整体而言，效率是非常高的。

另外加上一点，就是打基础和搜索问题。一个项目的开发可能需要用到一门新的语言或者软件，如果等我们学完所有东西再开发，那么效率是及其低下的，如果基础没打好就开发，会常常卡bug，同时编写的代码会非常乱，没有正确领悟到相关语言的优点，比如其中的布局元素，因此，我们需要先掌握好该语言或者软件的使用框架，利用这个框架，在开发的过程中，根据新的需求再延展出新的用法；遇到Bug网上搜索时，尽量查找官网，这样可以体系化的学习某一个控件或者事件，完善我们的知识体系。

- **表示层：收集用户请求并显示处理结果（UI, User Interface）**
- **控制层：拦截请求、调用业务逻辑处理请求、控制转向（UI）**
- **业务逻辑层：实现业务逻辑（BLL (Business Logical Layer) 、Service）**
- **数据访问层：数据的增删改查等（DAL, Data Access Layer）**
- **数据持久层：负责把数据进行持久化**

