小学期报告总结

此次小学期用TDD开发方法进行程序设计、实现、测试和交付。一开始，看了一本关于TDD开发的书，写出了需求一的测试用例，并先规定好要有视图类（Pos\_View）、控制类（Pos\_Control）和实体类（Pos\_Entity\_DataBase）。

在编写测试用例时，假设需求一所有类及类中成员变量及方法已全部完成，然后编写一个需求一整体运行的程序，然后输入什么，检查输出的是否合乎规定和期望。根据测试用例中的类名、方法名、变量名进行编程，也在这个过程中体会到内聚和耦合，发现有时还可以解耦合（实体类和控制类之间，视图类和控制类之间，有些参数耦合可以成为成员变量耦合，有的甚至直接耦合消失，这样更便于管理和回滚）。也让编程过程目标更加明确。

不可运行、可运行、重构优化。但是在最后一次检查，肖学长问我几个问题时才知道，程序编纂虽然完成，但是对开发方法——TDD 的了解还是皮毛。测试用例不仅要有输入输出的管理，还要再输入的时候考虑各个情况，比如会员积分的边界值测试（虽然编程时考虑到了，但需要有相应的测试用例）、同类商品不同价格、买二赠一和会员优惠不可同时获得等等。完成这些测试后，即可交付。

回滚很重要，多次迭代有不同的输入要求和输出要求，在需求之间的回溯和前进都要求程序方法内聚，分工明确，易修改，逻辑清晰。如果发现当前程序不符合当前需求，就可以回滚到上一个需求下进行重新编程。

在此感谢我的组员黄鑫宇、刘博洋，还有肖学长，舍友黄明明，是他们让我懂得如何使用trello、github和TDD编程方法，谢谢！

在使用trello的过程中，最开始我们理解错了trello的使用方法，错误的把每一个功能需求写成了一列，在第一次查的时候，助教告诉了我们错误的地方，并给我们展示了正确的使用方法，然后我们根据助教的提示将我们组的trello重新做了一遍，将每一个需求做成一张卡片，在清单中为组员分配工作，每一次迭代都在trello上写清楚步骤。

在github的使用过程中，最开始我们不会用git，在安装完git后我们不会使用，在网上查了教程后，可是可能因为版本问题，总是无法连接到我们的项目上，创建分支总是失败。然后再同学的指点下安装了github这个管理软件，通过这个软件可以更加的形象的管理我们的分支，更方便的上传版本，在不断地摸索中终于熟悉了这个软件，在上传github 的过程中，我们因为不熟悉敏捷开发的过程，在创建分支的过程中，出现了错误，通过组员间的交流，我们认识到了错误，将我们错误的分支删除，创建新的分支，并将每个人的代码进行了上传。

满足用户不断变化的需求是软件开发的长期无法解决的难题之一,经典的瀑布模式在一个迭代周期内表现优异,但一旦需求变化,瀑布模式却显得无能为力。敏捷方法满足需求的办法主要通过迭代。在每一次迭代周期结束时,都能交付用户一个可用的、可部署的系统,用户使用并体验该系统并反馈意见,在随后的迭代周期这些意见和需求的其他变化一起在产品中实现和集成。每次迭代周期应尽可能短,以便能及时地处理需求变化和用户反馈。

我们在实践的过程，并没有把用户的需求放在首位去重视，后来在助教的讲解下，明白了敏捷开发的真正意义，对一个功能进行测试，软件一直是可用状态。

在开始编码之前画一个草图、开发一个粗率的原型或者制作一些索引，都能提高你的生产效率。高效的开发者在编码之前都要进行建模工作。另外，建模是一种很好的在项目组成员与项目负责人之间沟通途径。你们在这个过程中探讨问题，从而对所要的是一个什么样的东西可以得到更好的理解，涉及到该项目中的每个成员也可得到对该项目有一个充分的了解。

组长：14080114 何焕野

组员：14080122 刘博洋

14080118 黄鑫宇

14080115 张明瀚