软件工程导论

小学期总结报告

组号：六组

成员：14080006 李佳龙

14080004 康宇晨

14080007 宋凯

14080003 秦心远

2016.6.30

**一、实践背景**

本课程设计是结合软件工程、程序设计开发、敏捷实践，软件测试，持续集成等课程的一个综合性设计课程。

通过本次课程，旨在让学生熟悉敏捷迭代开发方式，体会迭代式的需求管理应对变化时的优势。同时加深学生面向对象的程序设计和软件测试的理解。

本次设计采用分小组开发，实施敏捷开发过程，通过一个统一的案例需求，利用两周的时间完成该案例的多次迭代，并最终交付产品。

完成一个用于商场收银的POS机，整个开发过程会模拟真实项目的需求变化，需求会分为三个阶段逐步给出。

约束：

1）使用Intellij IDEA软件进行开发

2）使用JAVA语言进行编写

**二、需求总结**

**商品扫描**

POS系统扫描商品列表文件，读取其中所有的商品条形码。

**商品索引**

POS系统根据已读取的条形码信息，在商品索引表中搜索对应商品信息，并根据已读取的相同条形码个数，形成对应商品的数量

**总价计算**

POS系统根据商品信息和数量，计算商品总价

**打折**

POS系统根据读取到的商品折扣，在计算商品总价时计算折扣价以及该折扣节省的钱

**优惠**

对于有优惠活动标签的商品，POS系统要判断商品是只有优惠活动标签，还是既有折扣又有优惠活动标签，对购买两个以上带有优惠标签的商品，赠送一件，并且折扣和优惠活动不叠加

**会员机制**

会员基本资料存储在会员信息文件中（包括会员编号，会员名字，会员状态），扫描商品时输入会员编号，POS系统要扫描该会员编号下的商品列表，并在会员信息文件中读取对应的会员信息

**会员折扣**

商品索引文件中，增加会员专有折扣vipdiscount，对于会员状态为true的，可享受与普通折扣叠加的会员折扣，但会员折扣不和买二赠一优惠活动叠加

**会员积分**

会员信息文件中添加会员积分项，会员购物时，会参与兑换积分活动：

|  |  |
| --- | --- |
| 积分区间 | 积分规则 |
| 0 <=当前积分<= 200 | 每5元积1分，不足5元部分不积分 |
| 0< 当前积分<= 500 | 每5元积3分，不足5元部分不积分 |
| 500<当前积分 | 每5元积5分，不足5元部分不积分 |

**清单打印**

清单内容包括打印时间，所购商品名称，数量，单价，小计，总价，节省，优惠活动赠送商品名称和数量，会员编号，兑换得到的会员积分（对于非会员，清单不包含这两项）

示例：

\*\*\*商店购物清单\*\*\*

会员编号： USER001 会员积分：20分

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

打印时间：2015年01月04日 08:09:05

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

名称：可口可乐，数量：1瓶，单价：3.00(元)，小计：2.70(元)

名称：雪碧，数量：2瓶，单价：3.00(元)，小计：4.56(元)

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

总计：7.26(元)

节省：1.74(元)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**商品索引文件：**

{

'ITEM000000': {

name: '可口可乐',

unit: '瓶',

price: 2.00

discount：1.00

promotion：true

},

'ITEM000001': {

name: '雪碧',

unit: '瓶',

price: 3.00

discount：0.8

promotion：true

},

'ITEM000004': {

name: '电池',

unit: '个',

price: 2.00

discount: 0.8

promotion：true

}

}

**商品列表文件**

{

'user': 'USER001',

'items': [

'ITEM000000',

'ITEM000000',

'ITEM000000',

'ITEM000000',

'ITEM000000',

'ITEM000001',

'ITEM000001',

'ITEM000004'

]

}

**会员信息文件**

{

'USER0001': {

"name": 'USER 001',

"isVip": true

},

'USER0002': {

"name": 'USER 002',

"isVip": false

},

'USER0003': {

"name": 'USER 003',

"isVip": true

}

}

**三、Trello&Github**

Trello**：**

Trello是一个方便团队协作的工具，不仅是软件开发，任何团队协作都可以用trello来规划。在这次软件工程小学期里，我创建了“软件工程小学期”面板，并创建了to do,doing和done三个列表，用来记录我要完成的项目。首先是在to do列表中创建需求01-06，为每个卡片编辑需求描述和要求，规定是小组中的那些成员来完成这个需求，设置需求完成的截止日期，并将实现需求过程中要实现的功能、约束以及要测试内容在清单中列出来，在将该卡片编辑完成后，移动到to do列表中。在卡片设置完成后，手机的app会自动通知组员。在组员按照要求实现需求之后，在卡片的清单中的每一项左侧打上对勾，表示任务完成了，之后app会提醒我该需求完成，我就把卡片移动到done列表中，这个需求就算是完成了。由于组员都是一个宿舍的，这次trello的体验效果不是特别明显，但还是能从这次体验中，感受到trello的作用。Trello的规划能力很强，一张卡片就能将做什么，谁来做，截止到何时这几方面表述得很清楚，团队人员看到卡片上的描述和清单，能清楚地认识到自己的任务，哪部分该自己做，哪部分不需要自己做。同时，团长通过查看清单上所列出的项目哪些打勾，哪些没打勾，也能了解任务的当前进度，以便于计划以后的事。

Github：

Github可以和trello协作使用。组员在实现自己份内的需求后，在github上的团队仓库中，创建自己部分的项目分支，并将代码传到该分支上。这样，在每个组员都完成自己的任务后，分别将代码传送到对应的分支上，最后由管理人整合各分支上的代码，并传送到仓库的master branch上，这样整个项目就算是完成了。如果之后还会有新的需求或者需求变更，在完成trello上的清单后，组员还可以创建新的分支，作为更新的需求，再由管理人整合到master branch上，实现该软件的变更。

**四、敏捷开发**

本次设计采用了敏捷迭代开发方式，通过一个统一的案例需求，在两周的时间内完成案例的多次迭代，并交付最终产品。像这次的需求02和需求03，需求02实现了商品打折的功能，需求03实现了优惠活动买二赠一的功能，在单独测试02和03后，再将二者整合，这时需要解决的就只是接口，以及折扣和优惠不叠加。通过这次实践，我体会到了敏捷开发的优势。在每次迭代的过程中，虽然只完成部分需求，但可以直接运行并测试程序，整个POS机案例，被分解成六个需求，六个需求从最简单的打印清单、计算总价，到优惠活动，再到比较复杂的会员编号、会员打折，整个过程是循序渐进，逐步迭代将需求实现的。而且每个组员所实现的需求（子项目），都是可以独立运行，独立进行测试的，整个过程具备可视、可集成和可运行使用的特征。很多程序的bug和错误，算法的缺陷，在单元测试时就可以被发现和解决，这样在整合成最终程序后，需要解决的往往只是子项目接口的问题。与瀑布式开发这类传统开发方式相比较，敏捷开发是以迭代方式推进的，而瀑布模型式是最典型的预见性的方法，严格遵循预先计划的需求分析，设计，编码，集成，测试，维护的步骤顺序进行，步骤成果作为衡量进度的方法，例如需求规格，设计文档，测试计划和代码审阅等等。如果脱离小学期设计，升高到实际的软件开发，敏捷开发的软件一直处于可使用状态，它将项目分成若干相互联系、可以独立运行的子项目，因此，每个阶段都是可见的，客户可以直观的体验并提出需求变更。但瀑布式开发，客户可能在签了合同一段时间后，可以看见的只是需求文档，设计文档这些东西，不能清楚地了解到软件的进度和状态，在这种情况下，如果再提出需求变更，是很麻烦的。

**五、测试**

在三次迭代中，都使用了断言测试，要检验需求是否能够实现。测试的内容很多，要尽可能覆盖大部分情况。例如需求03的测试，内容有：

1. 购买单个参与优惠活动 具有折扣的商品
2. 购买单个参与优惠活动 不具有折扣的商品
3. 购买多个相同参与优惠活动 具有折扣的商品
4. 购买多个相同参与优惠活动 不具有折扣的商品
5. 购买多个不同商品 并且有的商品参与优惠活动 有折扣；有的商品参与优惠活动无折扣；有的二者含其一；有的二者都不含

像这样，才能真正尽可能地测试需求的实现程度，测试需求实现的缺陷。小学期这次因为程序不算太复杂，测试起来的内容也比较简单，但如果上升到真正的软件开发，那应该就复杂的多了。像之前老师上课提到的划分等价类方法，边界值分析方法，就用得上了。

**六、心得体会**

之前导论课上，老师讲的都是概念类的东西，很多都非常抽象，像敏捷开发、单元测试、集成测试，就算看了定义，看了例子，说实话对这些东西的印象也不是很深。因为在此之前，c程序，c++程序和java程序都是以个人为单位写的，对于团队协作这个东西也没体验过。通过这次小学期，总算是对这些事物都有了一定程度上的体现和进一步的了解。像敏捷开发的可视性、子项目可使用性，集成测试的接口，测试覆盖率这些，可以说都亲身体验了一次，让我们对软件工程又有了新的认识。之前一直觉得软件就是一个人写代码写出来的，现在看来没这么简单，之前做出来的都只是写小程序，根本算不上软件。真正的软件是有规划的，有团队协作的，有商业目的的，这方面我们要学习的东西太多了。同时，我组所有人都意识到了个人编程能力对于一个团队项目进度的影响，如果每个人的编程能力都很强，那么团队协作起来效率也高。因此我们还要学习很多东西，提高软件知识水平。