**网上酒店客房预定系统**

**项目评价报告**

南京大学软件学院

成员：叶晓波、辛志庭、田贵松、陈步兵

2016年12月31日

1. 概述

网上酒店客房预定系统的开发历时近三个月左右。这是我们的第一次大规模软件开发活动，也是第44小组的第一次合作。在开发过程中我们遇到了一系列的问题，当然也克服了一系列的困难，最终完成了本项目的开发。产品的质量基本符合需求，出彩的地方在于响应时间短，运行开销小；而用户界面和整体架构还有很大的提升空间。

1. 项目评审
   1. 项目所使用的过程是什么

按照课程的要求我们采用了瀑布模型进行项目开发，执行了明确的开发基线。

* 1. 实际的过程和原先确定的过程有什么不同？

大致相同，在部分阶段有返工的现象，如详细设计阶段对体系结构阶段的产品进行了很大的改动，包括新增控制器，重写桩和驱动程序等，以及构造阶段对人际交互阶段的界面设计提出了很大的改动。整个开发过程中对于需求的理解和实现也有很大的变迁。不过大体上还是按照我们预想的方向发展。

* 1. 进度表是如何随着时间变化和改变的？

进度表主要是按照课程设置的时间线来安排的，每周都有一个DeadLine催促我们工作。比较密集的开发时间是在每周的最后三天，提交产品的时间差不多也都在DDL前20分钟。有限的开发时间和紧凑的进度安排很大程度上影响了产品的质量。

* 1. 有多少个同步点和里程碑按时达到或错失？

几乎全部都按时达到了。但是在构造阶段的集成测试证明一直没有提交。

* 1. 过程的哪些部分运行的好？

好的部分在于前半段需求分析和最后的构造，大家效率都比较高，集会讨论的也比较频繁，团队交流比较密切。这很大程度上是得益于这两个部分对课程理论的学习比较深入，时间也比较充裕。

需求分析阶段我们曾经花费了很多时间探讨如何使产品在真实的使用环境下更好地满足客户的需求，从各个方面考虑了用户不同层次的需求；正因为如此我们认为到实现需求是一个合格产品必须满足的条件，需求实现的越多，用户体验越好，产品才越会被认为是成功的。

构造阶段我们重新回归到初始的需求，按照实际需求开发，剔除了一些不符合实际使用环境的功能，重新审视从前的文档和设计；更重要的是我们学会了灵活应变，根据构造过程的实际情况调整我们的工作安排，对一些重要的决策采用了更为灵活、更具有可修改性的方案。对于构造过程中发现的一些缺陷，本着实现需求的原则尽可能地去修正。自然，这个过程是我们能力提升最为显著的过程，没有之一。

* 1. 过程的哪些部分应该运行的更好？

做得不好的应该是设计阶段，主要原因是小组四人在这方面的经验都是最少的，对于课堂的理论知识的把握也不熟练，很多时候我们不能明确现在做什么是对的，项目到底需要怎样的架构方式。后来我们才意识到，软件架构这个领域没有绝对的对错，每种设计必然有其优点和不足，我们只要选择自己理解最为透彻的架构方式即可。

这个过程我们的返工是最多的，确定接口的过程极为痛苦，一个接口往往要商讨多次才能确定；甚至有一些接口一直等到构造阶段需要去实现的时候才有定论。

另外，详细设计阶段的大部分时间都用来修正体系结构阶段的历史遗留问题，对于底层的设计没有个=更多的关注，所以构造阶段就显得尤其痛苦——面临着很大的未知。

综上，我们在设计的方面还是需要多加学习。

* 1. 工具支持这个过程吗？

项目所采用的版本控制工具是Git，大体上支持这个过程。不过刚开始使用的时候在代码整合上吃了不少的苦头

* 1. 从整体上讲，这个过程运行的有效吗？

有效；我们觉得对于经验不足的开发者而言，瀑布的开发过程是比较容易上手的，但是如果想要让项目的质量进一步提高，必须采用螺旋模型。

* 1. 在今后，尤其要对哪些方面进行改进？

集成测试方面，对于maven等工具的掌控和理解；设计部分，需要对设计的每一个步骤有比较深刻的理解

* 1. 在每个阶段的每项任务上花费的时间是多少？

需求分析：4周

设计阶段：6周

构造阶段：6周

测试阶段：3周

部分阶段有重叠

总开发时间：16周

* 1. 在项目的上面周期中，产品是如何变化的？

产品在设计阶段末尾的时候就产生了大体的框架，但是在之后的构造阶段发生了较大的变更。最终的产品在测试阶段结束的时候才得到

* 1. 有没有出现重要的产品返工的情况？如果有，在什么时候？

有，如前所述，在详细设计阶段。

* 1. 工具支持产品的制造、维护和测量么？

支持。

* 1. 产品最后的规模有多大？

代码总量在20000行左右，其中75%集中在客户端，绝大多数的代码都属于ui层

* 1. 产品的质量如何？

一般，基本上实现了需求，但是可修改性和用户的友好度不够，也没有进行长时间的运行，其坚固性暂时无法验证

* 1. 团队（个人）工作中哪些风险发生了，其影响又是怎样的？

最大的风险在于设计阶段大家不知道该怎么设计，设计什么，该设计成什么样。其影响是十分深远的，甚至影响到了构造阶段的整体进度（整整多出了2周时间）。但是我们依然按时完成了项目的开发，当然质量因此也有些降低。

* 1. 在何时做出了哪项重要决定？这个决定又是如何影响这个项目的？

比较重要的决定在开始时选择所采用的技术：数据库MySQL、界面JavaFX、网络RMI、版本控制Git，并且没有准备采用任何新技术。影响不必多说，整个项目都是基于这些技术运行的。

* 1. 所遇到的主要问题是什么？

设计问题，对于一些设计模型中的概念理解不够（如VO、PO、DAO、Controller等，课堂上老师并没有给出详细的解释，书本上也语焉不详）；正是理论知识的掌握不足才导致了设计阶段麻烦重重，延误了时间。

* 1. 对这些问题的解决方法产生了什么效果？

解决方法是多看书，仔细阅读老师发来的demo，通过多读代码来加深自己对与这些设计上的概念的理解。至于效果，只能说亡羊补牢，为时未晚。

* 1. 开发团队成员是如何看待自己的职责的？

团队在前期的分工是不明确的，四个人分别负责四种用户的需求和设计。后期我们根据开发的要求更改了职责，两人负责界面设计，两人负责逻辑和数据库。而最后的情况是大家或多或少都会涉及到本不属于自己职责的部分。对于这种小规模的开发，而且还是初次尝试，我们认为理所应当所有部分都去尝试和了解一遍；而当我们逐渐成长为合格的开发者之后，坚守自己的职责，努力做好自己分内的事情，才是项目的根基所在。