1. **概述**

对快递公司管理系统的开发，从需求开始到最终项目完成，历时三个月时间，最终形成了可用的系统，并且这为以后的项目开发和团队合作积累了经验。从过程角度来讲，显示了很多问题，期待未来可以改进。

1. **项目评审**
2. **项目所使用的过程是什么？**

按照课程要求，使用瀑布模型。中间穿插了持续集成和测试驱动开发。后期以持续集成为主。

1. **实际的过程和原先确定的过程有什么不同？**

实际上的过程基本是按照教材的过程，没有什么大的变化。

1. **进度表是如何随着时间变化和改变的？**

如果按照Deadline算的话，基本没有什么变化。问题在于，小组自己设置的里程碑并不明确，主要以作业的Deadline为主。

1. **有多少个同步点和里程碑按时达到或错失？**

集中出现在项目的详细设计阶段。原因如下：1、没有充分理解领域模型的作用，所以构造的时候还需要重新考虑，造成返工。2、接口脆弱，定义不明确，所以构造的时候还需要重新考虑，造成返工。

1. **过程的哪些部分运行的好？**

持续集成做的非常好，需求分析做的非常好，逻辑和数据完成较快缺陷较少。

1. **过程的哪些部分应该运行的更好？**

人机交互设计启动太慢，界面没有足够的时间来完全美化。详细设计的接口应该定义地更加明确。

1. **工具支持这个过程吗？**

没有使用重量级的过程工具，持续集成的一系列工具可以正常使用，度量的一些轻量级工具也可以工作。

1. **从整体上讲，这个过程运行的有效吗？**

对于没有经验的开发者来说，这些过程非常重要，帮我们解决了程序开发复杂度过高的问题，用工程化的思想培养我们，帮助工程最后成功收尾，所以整体是有效的。

1. **在今后，尤其要对哪些方面进行改进？**

文档编写时成员之间的交流要加强，风格要统一。详细设计一定要详细，要设计稳定的接口，要使用OO的思想来进行逻辑层的详细设计。

1. **在每个阶段的每项任务上花费的时间是多少？**

需求开发：2015.9.16-2015.10.14

软件设计：2015.10.15-2015.11.17（体系结构设计和详细设计各占一半）

软件实现：2015.11.18-2015.12.09

软件测试：2015.12.3-2015.12.20

软件交付：2015.12.21-2015.12.26

软件维护：未至。

1. **在项目的上面周期中，产品是如何变化的？**

产品逐渐成形，这个问题也许对迭代式的模型更有意义

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。在构造阶段修改了体系结构设计中定义的部分层与层之间的接口和详细设计中的很多关键类的接口。

1. **工具支持产品的制造、维护和测量么？**

工具使用量是大大不足的 ，使用的部分支持测试、度量和集成构建

1. **产品最后的规模有多大？**

LOC：30,000

1. **产品的质量如何？**

产品的质量被认为达到预期，详见总测试报告。

1. **团队（个人）工作中哪些风险发生了，其影响又是怎样的？**
2. 分工不合理，中期对界面的工作量预估错误，导致后期界面跟不上进度。
3. 存在不遵守团队规范的行为，预定的惩罚没有执行，总归会影响到氛围和心情。
4. 存在特殊的原因拖慢项目的进度，比如成员电脑坏了，有必须参加的活动。
5. **在何时做出了哪项重要决定？**
6. 构造阶段决定修改定义好的部分接口
7. 构造阶段决定修改分工。
8. 使用设计模式重构代码
9. **这个决定又是如何影响这个项目的？**
10. 有助于实现用户需求和业务目标。
11. 方便实现，但是影响了进度。
12. 加快项目的进行。
13. 增加了代码的可维护性。
14. **所遇到的主要问题是什么？**

最主要的问题在于，每个阶段都需要对上个阶段甚至以前阶段的制品进行修改，尤其是涉及阶段的制品，详细设计最初稿有不少都是无效的，这是工作量不断增大的一个很重要的原因。

同时设计的质量也阻碍了构造阶段的进行。

任务分工不合理，导致进度时有滞缓。

1. **对这些问题的解决方法产生了什么效果？**

重新明确成员职责，项目进度加快，并且使我们维护代码的成本降低。

1. **开发团队成员是如何看待自己的职责的？**

总得来说，团队成员都很负责，能够认真完成分配的任务，对项目有一定的责任感。