

景德镇陶瓷大学

工学学士学位论文（设计）

题目：校园 O2O 便捷式商城建设

学 号： 201210510118

姓 名： 付栖凤

院 (系)： 信息工程学院

专 业： 12 计算机科学与技术

完成日期： 2016 年 5 月 15 日

指导老师： 胡开华

分类号_____

学校代码_____

UDC _____ 密级 _____

学 号 201210510118

景德镇陶瓷大学
信息工程学院毕业论文
校园 o2o 便捷式商城建设

付栖凤

指导老师： 胡开华 讲师 景德镇陶瓷大学 信息工程学院

申请学位级别 学士 专业名称 12 计算机科学与技术

论文提交日期 2016 年 5 月 17 日

论文答辩日期 2016 年 5 月 22 日

学位授予单位和日期 景德镇陶瓷大学

答辩委员会主席： 肖绚

论文评阅人： _____

2016 年 5 月 15 日

目 录

摘 要.....	V
Abstract.....	错误!未定义书签。
目 录.....	III
摘 要.....	V
Abstract.....	VI
第 1 章 绪论.....	1
1.1 研究背景.....	1
1.2 研究现状.....	1
1.3 本文目的.....	2
1.4 本文创新.....	2
1.5 本文贡献.....	2
1.6 本文结构.....	3
第 2 章 项目总括.....	4
2.1 项目并发进程.....	4
2.2 特定领域软件架构.....	5
2.3 从业务建模步入系统开发生命周期。.....	5
2.4 集成设计工具 Enterprise Architecture	6
2.5 集成开发工具 IntelliJ Idea.....	6
2.6 本章小结.....	7
第 3 章 项目规划.....	8
3.1 项目启动风险.....	8
3.2 电子商务应用调研.....	9
3.3 Java 平台概述	10
3.4 项目经济可行性分析.....	10
3.6 系统决策.....	11
3.7 本章小结.....	12
第 4 章 系统分析与设计	13
4.1 概念建模与分析.....	13
4.1.1 原型系统设计.....	13
4.1.2 用例建模.....	14
4.2 系统模块的设计.....	18
4.2.1 WebApp 设计	18
4.2.2 数据库设计.....	19
4.2.3 Struts2 模块设计	23

4.2.4 Spring4 模块设计	24
4.2.5 Hibernate 的 DAO 模块设计	24
4.2.6 Hibernate 的 Model 设计	25
4.3 本章小结	26
第 5 章 系统编码与实现	27
5.1 系统任务分解	27
5.2 项目估算	28
5.3 软件开发策略	29
5.4 子项目 MyPro	30
5.4.1 MyPro 的系统分析与设计	30
5.4.2 MyPro 的数据库设计	35
5.4.3 MyPro 系统编码与实现	36
5.5 子项目 Shop	39
5.6 本章小节	42
第 6 章 文档编制	43
结束语	44
致谢	45
参考文献	46

摘 要

据国家统计局统计，2015 年普通高校在校学生达 2804 万人，数量庞大而且稳定的电子商务消费群体催生了许多针对大学生的校园 O2O 电商项目。另外，目前众多企业的 SSH2 系统由于技术老化、软件施工不规范等因素，导致后期维护困难。许多教材仅仅针对软件开发生命周期中的一部分进行描述，而无法给出整个项目从需求分析到代码实现的详细过程，难以达到人才培养目的。

因此，本文以 O2O 电子商务作为应用领域、SSH2 作为项目的软件架构、SDLC 作为软件开发模型，进行项目的施工。最后项目在功能实现方面完成了对 Supplier、Address、Customer、Mytable 等表的增删改，普通查询及元数据查询；业务实现方面完成了顾客和供应商的注册、登录、信息修改，商品从选购到生成订单这一简单业务流程。

关键词：020 商城 SSH2 软件架构 项目估算

Abstract

Based on data from National Bureau of Statistics of China which show that there are 28,04 million students on campus. And so large scale and so stable numbers of consumers in e-shop that delivered many of O2O application. On the other hand, So many of SSH2 system hard to maintain because of technology is being aging and unstandard of software process model. What's more, so many of teaching material just lightly described the fragment of SDLC and could not give the project which start from requirements analysis to code make system come true, it to be far inferior training more excellent talent.

So the article makes O2O e-shop as application domain, makes SSH2 as project's software architecture, make SDLC as software process model, go for developing project. In the end, about functional, project finished CRUD of database table named Supplier, Address, Customer, Mytable and so on. and an metadata query function also finished in *MyPro*. About business requirement, project finished customer and supplier for login and modify their account info. Serial of function about *shopping process* also finished.

Key words:

O2O e-shop SSH2 software architecture project estimation

第1章绪论

1.1 研究背景

随着经济的快速发展，O2O 电子商务越来越受到人们的关注。各种线上线下结合的服务已如雨后春笋般出现在各大城市，大学生作为一类重要的消费群体，占据了电子商务消费者的很大一部分。同时，随着全国经济的不断增长，人们的物质、精神、文化生活水平不断提高，学生对于电子商务模式多样性的要求也越来越高，更多的需求都将刺激 O2O 电子商务的发展。因此，面向主学生消费群体的 O2O 设计有着良好的现实意义。

基于 SSH2 的大型系统已在企业取得良好的市场，通过系统剪裁，可以完胜不同的企业应用需求，即可以接入 web 应用，也可以接入 java Swing 桌面系统，通过 websocket 协议，甚至能接入基于 html5 移动客户端，基于 SSH2 的系统开发与研究过程，会令参与者更深刻体会到软件工程领域的各种理论，许多高校引入它作为教学内容。

软件工程理论在软件项目中的体现与运用十分普遍，然而众多的理论与完整的软件生命周期相结合的描述却十分不是很多，因此在一定程度上阻碍了领域优秀人才的培养。本文所涉及的“O2O”项目是零起点项目，作为这些理论的载体进行了进行场景演绎与描述。

1.2 研究现状

(1) 国内的软件领域，软件工程理论就针对系统开发生命周期（SDLC）中的某一阶段进行系统模型建立，或者针对开发方法与技术细节进行研究，并给出示例；而软件项目，则直接给出项目代码，象征性地给出软件工程上游产物与理论。无论是理论研究为目的的“以涵盖 SDLC 的项目作为载体进行全面的理论阐述”，还是以项目开发为目的“项目贯穿系统建模、用例建模、软件架构设计、项目估算、项目任务分解、程序设计等理论”，两种“场景”在国内比较少见。

(2) 项目估算理论在国外已十分成熟，比如规模估算，一种是利用“项目规划”阶段所花的时间，来进行项目估算整个项目工期，另一种则是采用用例点估算。然而项目估算在案例中的体现，却在书本上踪迹难觅。

(3) 目前大多数的 SSH2 系统很少有大量使用注解，以此减少资源配置、减少软件开发工作量。

1.3 本文目的

本文目的是通过以“校园 O2O 便捷式商城建设”项目开发为依托，进行系统演绎，尝试：

- (1) 开发一个校园 O2O 便捷式商城应用。
- (2) 以校园 O2O 便捷式商城建设为载体，设计一个可复用的软件架构。
- (3) 对项目估算理论的应用进行探索。
- (4) 开发一个标准工程 MyPro 用于定义项目技术要求、代码风格、开发流程等。项目完成基本增删改查、基于元数据的查询。
- (5) 引入 struts2 约定，减少资源的配置工作量，对市面上大多数同型 SSH2 系统代码工程提出改进建议。
- (6) 选取一个拥有相同软件架构的 Shop 系统，作为参考系统和可能的代码复用来源，进行二次开发。

1.4 本文创新

阵位法系统工程。它从静态方面描述了本工程实施所需要的技能技术，基于五个知识域的确立，将项目所涉及的到方方面面纳入一个系统，建立本项目的系统认知。

菱形法软件施工。菱形法主要解释本系统动态进程，菱形上半部为设计过程，下半部为开发过程，上半部的顶点为原型界面，下半部下顶点是项目核心代码或者最佳参考模型。个人项目通过系统设计稿的完善、代码逐步实现完成到方形的过渡，菱形法主要从动态方面描述了工程的时间维。

SSH2 软件质量提高。基于一个测试用例建立了一个名为 MyPro 的 Web 项目，它给出了一种提高 SSH2 系统的软件质量的参考，从代码的工程结构、前台 jsp 页面代码、后台增删改查代码进行代码简化，从而提高代码的精练性，在查询方面，给出了一种基于元数据的关键词查询方法。

1.5 本文贡献

本文贡献主要为基于 MyPro 的项目，它可以以它的规范性和先进性对企业同型应用的开发提供指南，同时可作为教学项目进行讲解，亦或深入开发。原型系统方面也可以细化，成为一个良好的 Html5 项目。并可能接入 MyPro 或 ShowPro 系统当中，充当系统的移动客户端。

1.6 本文结构

本文一共包括 9 个章节。

第 1 章为绪论。主要介绍了本文背景、知识现状、本文目的、本文创新、本文贡献、本文结构，共 6 个小节。

第 2 章为项目总括。内容涉及项目并发进程、特定领域软件体系结构、从业务建模步入系统开发生命周期、Enterprise Architecture 及 IntelliJ idea 两种先进的工具，项目并发进程描述项目在施工过程中同时进行四个不同的大任务，其它内容分别为建立本文系统认知服务。

第 3 章为项目规划。本章主要讲了项目实施所采用的相关技术以及项目启动前做的一些活动。内容包括项目启动风险、电子商务应用调研、JavaEE 平台概述、系统决策、项目经济可行性分析等内容。

第 4 章为系统分析与设计。主要采用各种分析策略进行系统建模与设计工作，围绕原型系统开发、用例建模、数据库设计、软件架构设计等核心任务进行软件工程上游活动。

第 5 章为系统编码。基于前一章节所设计的数据库及软件架构进行初步的任务分解、项目规模估算等，并进行 MyPro 子项目的开发——用于建立开发标准，Shop 子项目的二次开发——进行代码复用、ShowPro 开发——目标系统功能实现。

第 6 章为文档建制。本章主要描述项目在开发过程是涉及到的主要文档产物，文档主要分项目组内部交付文档和客户交付文档、辅助说明文档等。

第2章 项目总括

2.1 项目并发进程

项目在开发过程中主要划分了四个开发进程：

系统分析与设计。系统设计主要利用 enterprise architect 系统集成设计工具进行，结合 visio 进行辅助设计，在系统设计过程中，也需要大量的编码，从而得到适合于构建系统的软件架构以及系统细节方面技术要求，本项目在设计过程中主要文档产物是 training.eap 和 showqiu.eap，系统设计偏向于从概念入手，在需求及任务模块不明确的情况下进行，目标系统的系统分析与设计过程可参考 MyPro 子项目的相应阶段。

系统编码与实现。主要利用 intellJ 进行.html、.jsp、.java、.sql 等各种文件的代码开发，用 myeclipse2015 的逆向工程实现系统的实体类、DAO 类以及资源配置文件的自动生成，这些方法将有效提高软件开发效率。intellJ idea 支持由代码生成 UML 图、抽取接口、可视化数据库设计等功能，这些都能极大地提高系编码效率。

文档拟制。文档建制主要涉及到软件开发过程中一系列说明文档，包括的产品相关文档如《系统请求》、《用例说明文档》、《软件需求规格说明书》、《软件设计说明书》、《数据库设计说明书》、《系统测试计划》等，标准规范文档如：文档标准、代码标准、数据设计规范等，还有基于相关调研文档、参考文档、项目管理控制文档等。

辅助过程。辅助主要涉及到项目培训资料、可复用组件选择、参考软件系统的代码复用、项目调研、项目资金人力等资源调配等。

基于上述过程，项目结束时，本项目进行打包交付的四大模块主要有：showqiu.eap，项目最终版本代码，项目涉及文档，参考（参考软件系统，备选组件，培训资料等）。

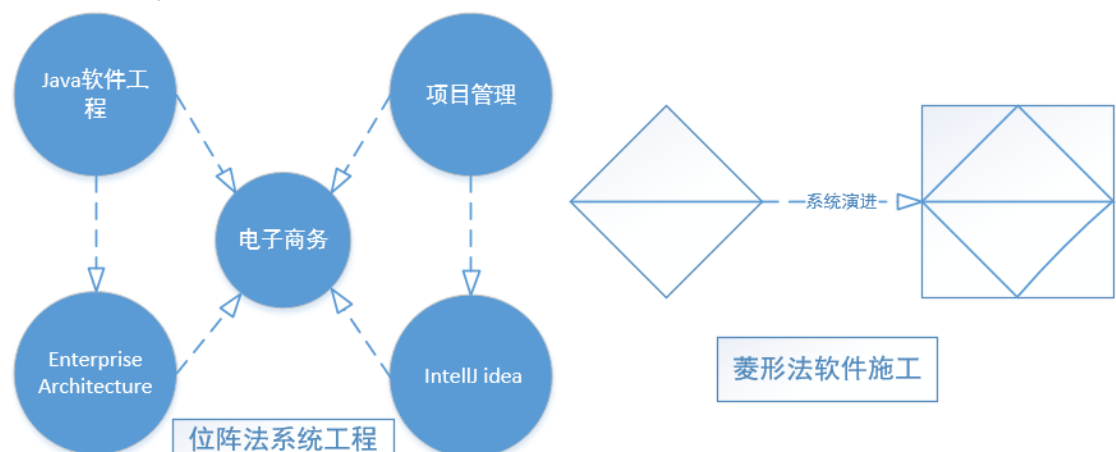


图 2.1 项目知识调用与动态开发抽象图

Fig2.1 organizing the knowledge project invoked ,and abstract of dynamic diagram developing

2.2 特定领域软件架构

软件体系结构又称为架构，在系统分析阶段，它发挥着巨大的作用，一方面，借助于软件体系结构进行描述，可以使问题得以进一步抽象，使得整个系统更易于被系统分析设计人员把握，更清晰地认识系统，完善对系统的理解。除此之外，它还为系统分析人员提供了新的思路。比如在更高层次上进行系统一致性检查、使用成熟的体系结构风格。另一方面，它能帮助软件系统的各有关权益方（客户、用户、项目管理人员、设计开发测试等人员）形成统一认识，互相交流。

特定领域的软件架构 DSSA(Domain Specific Software Architecture)的研究目的是为了软件复用，在一个领域中，不同系统及同一系统的不同版本之间在体系结构上存在着相似性，而现在市场上存在各种代码、组件可以复用，如果得到一个我们当前开发所需要的 DSSA，并且有大量的代码、组件可供复用，那么我们开发任何一个基于该领域的任何规模的系统或应用，我们都能够较快地进行。特定领域下的一个系统开发模型如双工模型，即用户与领域专家共同协作。

在本文所涉及的项目中，便进行了基于 SSH2 的系统架构设计，该架构是通过众多同型系统类比，取其精华后，得到的，并且，使用接口隔离来降低模块的耦合度，遵循高内聚，低耦合原则，具有了一定的先进性。目标系统 ShowPro 中更能体现此架构特点。

2.3 从业务建模步入系统开发生命周期。

业务建模的输出是业务模型（Business Model）。阶段工作主要是基于商业世界的分析，通过构建现行系统静态描述模型、分析现行系统动态运行机理、得出其现有弊病、尝试性构建愿景模型。业务建模方法推建采用 Eriksson-Penker Model 业务建模语言。具体详细参考理论综述类[04]。

业务建模中的重要一环又是数据建模，数据是企业应用的核心，数据建模的最终输出是成熟的数据库架构，我们可以采用建立系统原型界面、网络资源调研、问卷走访、与客户交流等方式进行。联合需求调研等方式进行需求捕获，当我们需要实现一个系统时，必然要掌握用户的数据处理信息化需求，在数据分析过程中通过概述实体确立、逻辑实体确立、属性确立、候选键确立、主键确立、外键确立、关系确立、基本表/视图确立、列确立等一系统列工作，同时，规范化与反规范化技术的使用也会体现其中，关于数据建模的详细细节可参考理论综述类[05]。书中给出的数据生命周期如下：

	数据需要	数据规划	收集、恢复	
删除数据	数据生命周期			存储
	存档	数据生产	整合	

图 2.2 数据生命周期示意图

Fig2.2 Data life cycle

利用业务建模来定义软件架构，即确定信息系统相关需求。从业务建模到软件架构到项目验收的实现过程图如下：

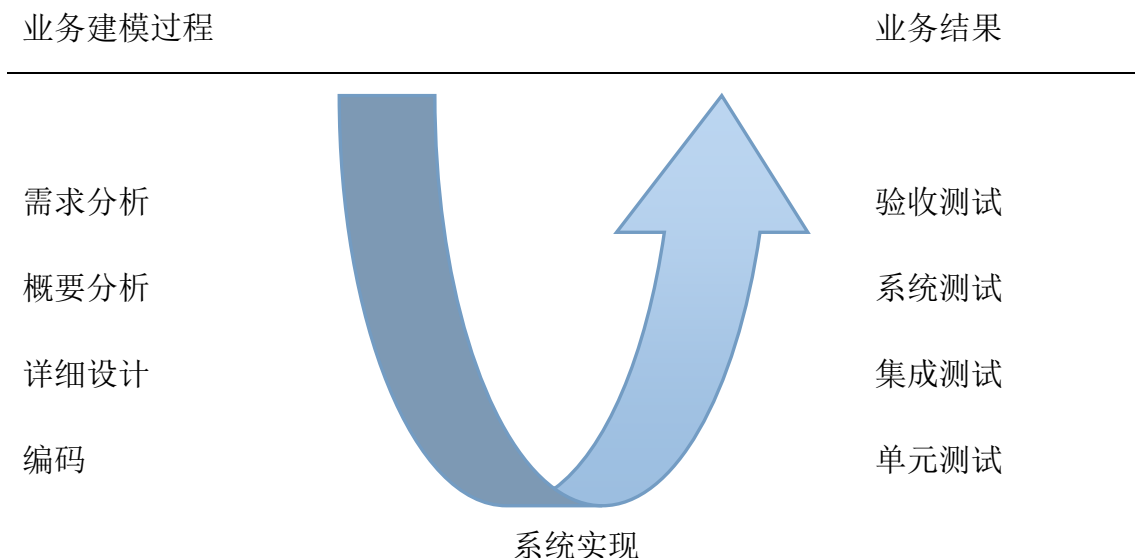


图 2.3 业务建模与系统开发周期的示意图

Fig2.3 Business and SDLC

通过领域分析、复用特定领域的领域模型，参照《用户需求规格说明书》并对商业世界进行业务建模，从而可以进一步得到系统的用例模型，用例模型是结合《用户需求规格说明书》中所定义的需求业进行“项目需求定义”活动的，用例建模图中包含数据模型和进程模型，基于用例建模进行用例规约，将得到《用例说明文档》，然后再选用相关组件，进行系统决策与架构模型设计。

2.4 集成设计工具 Enterprise Architecture

系统设计工具 EA（Enterprise Architecture）是一款最先进的系统设计工具，不仅集成了基本的 UML 类图、包图等图标拖曳绘制功能，还可在进行基于 Eriksson-Penker Model 方法的业务建模、Web 建模、项目管理等，基本涵盖系统工程所有建模功能。本项目所用的 EA 属于英文破解版，因此有些地方还是用简单的 visio 进行着相关图的绘制。

2.5 集成开发工具 IntelliJ Idea

IntelliJ IDEA 是 JetBrains 公司的产品，这家公司总部位于捷克共和国的首都布拉格，开发人员以严谨著称的东欧程序员为主。intelliJ Idea 是一款综合的 java 编程环境，它提供了一系列最实用的的工具组合：智能编码辅助和自动控制，支持 J2EE、Ant、JUnit、CVS、maven、各数据库设计开发等的集成，非平行的编码检查和创新的 GUI 设计器，并且，由于它是基于 java 的综合开发平台，所以从它身上也可以系统地学习到许多 java 领域的概念，下图是该平台所支持的开发工程。IDEA 把 Java 开发人员从一些耗时的常规工作中解放出来，显著地提高了开发效率。具有运行更快速，生成更好的代码；持续的重新设计和日常编码变

得更加简易，与其它工具的完美集成；很高的性价比等特点。虽然项目启动时开发人员对 IntelliJ 不熟，但掌握好它，对于未来项目的开发有良好的促进作用，基于以上，本项目的开发优先选择它。本文采用它进行数据库设计与开发。

2.6 本章小结

本章重点介绍了本项目中的一些重要理论及概念。在本文件所涉项目，从需求输入的开始，经历需求解析、应用调研、技术选型与路线制定等过程，到 MyPro 知识工程的实现的过程中，这些理论在指导本项目施工的同时，项目施工的各种现象与机理又与这些理论产生共通。

本章的核心（产物）是“项目知识组织结构与动态开发抽象图”，因为它从最高层次抽象地描述了本文项目的方方面面，强有力地描述。

第3章 项目规划

3.1 项目启动风险

(1) 应用的熟悉度

应用的熟悉度主要包括两部分，一是应用系统所服务的领域熟悉度，以及应用系统平台熟悉度，二是应用程序实现所涉及到的相关技术的熟悉度。

“校园 O2O 便捷式商城”属于“电子商务”应用领域，项目启动时，由于对电子商务应用和 Java 平台的不是很熟悉，因此对系统的实现是一个高风险因素，基于此，通过查阅相关资料，生成了一份《电子商务应用调研报告》参考文档，该文档概要地介绍了电子商务的一些基本概念，以及所涉及到的相关技术，该文档结合本项目的特征和开发者的知识技能，建议本电子商务平台开发平台采用“java EE”平台，并给出了该平台下的系统结构图和基本物理视图。《电子商务应用调研报告》降低本风险因素的主要解决方案。

(2) 技术的熟练度

“校园 O2O 便捷式商城建设”应用实现所需技术拟采用 Java Web 技术，而数据库平台采用 mysql，本文的技术熟练度所针对的项是原型系统开发技术、Jsp 技术、SSH2 框架的 Jar 包使用技术、数据库设计技术、系统建模技术、项目管理技术、java 编码技术等、注解使用技术。

(3) 项目规模及边界

第一次系统设计与开发，不清楚项目涉及用例个数、前后台编码量、数据库表的张数等，全部未知。唯一得到的信息是可以采用用例进行估算它的规模。采用用例估算规模时，用例的颗粒度、用例叙述的细腻度、技术环境的评价都会导致估算不准确。所以估算模型一般建立在大量的实践经验基础上的。

用例点=未调整用例点×技术复杂因子×环境因子
未调整用例点=×参与者总权重+用例总权重
技术复杂因子=01.4+(-0.03×环境总权重)
一个用例点大约需要耗费 20~28 人时 (ManHours)

基于以上，本项目首先开始进行技术培训、应用调研分析、先行试验性开发，并建立项目文档标准、代码风格标准等。

3.2 电子商务应用调研

电子商务应用调研的目的拟在建立起目标系统的功能需求、非功需求，消除开发者对系统的不熟悉，从而减小项目风险。基于调研得到的信息如下：

(1) 电子商务平台总体规划内容主要包括：确定网站建设目的、定位网站客户、理顺结构和层次、设定网站盈利模式、设定主要业务流程。

(2) 电子商务网站的基本组件有：应用服务器、工作流和群件子系统、内容管理子系统、目录服务器、性能优化工具、邮件和消息服务器、搜索引擎、网站服务器。

(3) 电子商务网站的基本功能主要有：信息发布、网上订购、网上支付、咨询洽谈、电子账户、服务传递、意见征询、交易管理。

(4) 电子商务安全体系如表下：

表 3.1 电子商务安全体系

Table 3.1 security system of e-commerce

电子商务安全体系	
应用层	保密性、完整性、可用性、可控性、不可否认性、微分可认性
安全层	SSL 协议、SET 协议
安全认证层	数字摘要、数字签名、认证中心……
加密技术层	对称加密、非对称加密
网络服务层	入侵检测技术、安全扫描、防火墙……

(5) 电子商务规划报告的内容包括：

- | | |
|------------------|---------------|
| ① 系统背景描述 | ⑧ 联机交易中的支付与认证 |
| ② 企业需求描述 | ⑨ 系统安全及管理 |
| ③ 电子商务系统设计的原则及目标 | ⑩ 系统性能优化及评估 |
| ④ 商务模型建议 | 11 系统集成方案 |
| ⑤ 目标系统的总体结构 | 12 系统开销与投资 |
| ⑥ 应用系统方案 | 13 实施方案 |
| ⑦ 网络基础设施 | 14 商务系统收益分析 |
| | 15 其它说明 |

大学网购，是以一定的地理区域为服务范围，以大学校园为代表的一种小而全的校园超市电子商务模式。它能现校园内实体店和网点之间的一种完美结合，是一种典型的 O2O(线上和线下) 校园电子商务模式。有以下三个优点：

(1) 市场大/ 针对大学校园人群密集区市场。市场大下一个亿万级市场的机

遇。业内人士预计：大学校园生活类 O2O 会是下一个亿万级市场，是目前市场上唯一能产生超级电商的领域。

(2) 易推广/ 目标人群固定，宣传方式简单效果好。易推广推广是做好电商最重要的一个关节，客户设定为大学校园人群，范围特定，易宣传易推广，短期推广就能带来很好的效果！

(3) 运营易/ 最少 2 人即可运营，投入少收入多。

3.3 Java 平台概述

电子商务系统平台的选择与配置中，平台主要有 .net 平台、php 平台、javaEE 平台。本文主要是基于 JavaEE 平台进行概述。

本文项目基于 java 平台的开发主要采用比较流行“Spring+Struts2+Hibernate”框架（即 SSH2，2 代表 struts2，struts1 与 struts2 完全不同），表现层的模板技术为 jsp(可选的决策选项还有 freemarker)，数据库层采用 mysql(其它选项有 oracle、sql server、mongodb 等，假设经决策讨论选定 mysql)。

SSH2 架构中，Struts2 的充当 Webapp 事件处理者，并对 Action 类的处理结果配置相应的视图，是前端 MVC 模型的控制层。本文的 MyPro 项目在 jsp 页面中大量采用 struts2 标签，比如通过标签完成表单数据的绑定，从而达到精简页面代码。

Spring 充当 SSH2 系统的核心管理者，管理 Struts2、Hibernate 以及 DAO，管理方法有两种，一种是基于配置的管理方法，这种方法有利于集中管理资源，采用切面，但大型系统类非常多时，大量的配置却又给系统维护带来一定困难；另一种是基于注解的管理方法，此方法不需要配置资源，直接开启注解扫描识别含有注解的类。

Hibernate 在 SSH2 当中，主要将数据库中的基本表及关系映射成实体类，及实体关系类。并为每个实体提供 DAO 类。

3.4 项目经济可行性分析

软件工程经济学与其它经济学分支又有所不同，作为技术人员，独立的项目设计与研发过程中，除了项目成功后，每推销一次可得若干资金外，解决领域问题的能力也将给技术人员以身价的提高（即可胜任更高工资水平的工作），而基于技术撰写的书籍或者论文，也将带来不菲的收益。假如本项目给定开发周期设为两年，项目投资资金回收期定为两年，可得出如下表 3.2。

表说明身价收益是指完成项目开发后，开发者凭借本项目积累的经验，能够承担市场年薪 144K 或 200K 的项目。三个资产集收益是指完成项目开发后，开发者应用该项目模式，以娴熟的技能能快速衍生出类似产品并投入市场。开发支出主要是开发者人力投入，电脑设备购买、办公场地租赁、项目完成后的宣传、

支持等工作。由于本项目含技术驱动开发、开发员技术熟练、项目周期长等存在高风险，固为了预防风险。在估算费用乘上系数 1.14，修正为 400K¥，当投入费用可能超 400K¥时，应当慎重考虑是否追加费用。

表 3.2 项目经济分析

Table 3.2 economic analysis of project

数值单位 (K¥)	2015	2016	2017	2018	合计
开发收益					
薪资定位收益	0	0	144	200	344
两个研究切入收益	0	0	20	40	60
Intellj+EA 技术实战书	0	0	100	100	200
系统开发模型	0	0	10	20	30
收益合计	0	0	274	360	634
NPV					
开发支出					
人员时间薪资无形损失	60	60	0	0	120
伙食费	4	4	5	5	18
办公室租赁	0	0	9.6	9.6	19.2
参考书籍	0	5	5	0	10
项目推广支持	0	0	4	4	8
风险防范	1	1	2	2	6
总支出	65	65	25.6	20.6	181.2
NVP					
投 资 回 报 率 (474/349.2-1)	249.9%	预规划:	10	风险线:	200

3.6 系统决策

在项目启动初始阶段，有对需求获取方法的决策，有对软件架构进行选型的决策，开发工具、开发方法、迭代策略、数据库平台、连接池……一系列决策项，在项目施工过程中，不断涌现，有的决策影响较小，比如后台查询到的信息保存在 list 中，通过 EL 表达式，还是 struts2 的<s:property/>标签显示，两种显示者可以，只不过代码精练程度不一样，而有的决策影响非常大，比如项目在使用 Struts2 之前，使用过 Struts（基本淘汰了），事实上它们是两种完全不同的框架，从而导致项目返工，可见，重大决策之前的调研非常重要，必要时还需写决策报告，记录选项的优劣等内容。

表 3.3 系统设计时考虑的若干选项

Table 3.3 several items may be thinking while designing the system

决策项	设计时决策选项				对比	决策结果
	A	B	C	D		
系统开发方法	瀑布模型	迭代模型	敏捷开发	增量开发	
平台选型	.net	php	javaEE		
表现层模型	jsp	freemark	velocity	beelt	
对象持久化 JPA	Hibernate	TopLink	OpenJPA		
数据库连接池	Dbcp	Proxool	C3p0		
数据库	关系型数据库系列		非关系型数据库系列		
	Mysql	Sql server	mongodb	DB xml		
应用服务器	tomcat	Glassfish	Jetty	Resin	
服务框架选择	Axis	XFire	CXF	
.....	

3.7 本章小结

在项目规划阶段,通过电子商务应用调研、平台技术调研与试验性编码,对降低项目风险起到了一定的作用,对拟进行的项目有了一个基本的认识,基于调研还为项目开发技能技术培训提供了清晰的任务安排,本文基于调研之后,便开始了 UML、HTML5、SSH2、Mysql 等的技术培训,还包括开发工具使用、开发环境搭建、JDK 和 Tomcat 及各种 Jar 版本的选择、项目文档模板建立,环境测试、试验性编码等工作。

第4章 系统分析与设计

4.1 概念建模与分析

4.1.1 原型系统设计

原型设计采用原型开发方法，原型法开发是从需求收集开始，开发者和客户在一起定义软件的总体目标，标识出已知的需求，并规划出需要进一步定义的区域。然后就是“快速设计”，快速设计集中于软件中那些以用户（或者客户）可见的部分的表示（如输入输出格式）。原型由用户评估并进一步精华待开发软件的需求，这个过程是迭代的。理想情况下，原型可以作为标识软件需求的一种机制，建立了可运行的原型，开发者就可以在其基础上试图利用已有的程序片断或使用工具（如报表生成器、窗口管理器）来尽快生成可运行的程序。因此本项目的这个模块完全可以通过消化网上的一些美观的同型系统，来构建目标系统。

基于图 4.1，本项目所设计的原型对目标系统的需求定义如下：

(1) 在 `index.html` 文件主要动作为顾客单击注册与登录，顾客进行以关键词、以商品分类、以价格段等方式进行商品搜索。并通过单击商品的查看详情进入到 `detail.html` 页面当中。

(2) 在 `detail.html` 主要功能定义为单击商品加入购物车或心愿清单，填写购买数量，填写配送地点，单击提交后转到 `cart.html` 页面。

(3) 在 `cart.html` 主要用于显示商品购物车中的内容，提供修改删除购物车中的商品条目，最后单击进入结算页面 `address.html`。

(4) 在 `address.html` 中，主要是填写配送地址、添加支付方式。最后单击提交订单。跳转到 `order.html`。

(5) 在 `order.html` 中主要显示订单信息。

(6) `index.html` 页面头部中，单击“账户”，可以查看或修改自己的订单信息，也可以查看自己的订单历史。单击“购物车”跳转到 `cart.html` 中查看购物车中的商品，单击“心愿单”同单击“购物车”，以上单击基于客户已登录，若未登录，则无效，单击“登录”跳转到 `login.html`，没有注册的话，单击 `login.html` 中的去注册，跳转到 `register.html`，完成注册后自动跳转到 `login.html` 界面，并自动填写注册时填写的信息。

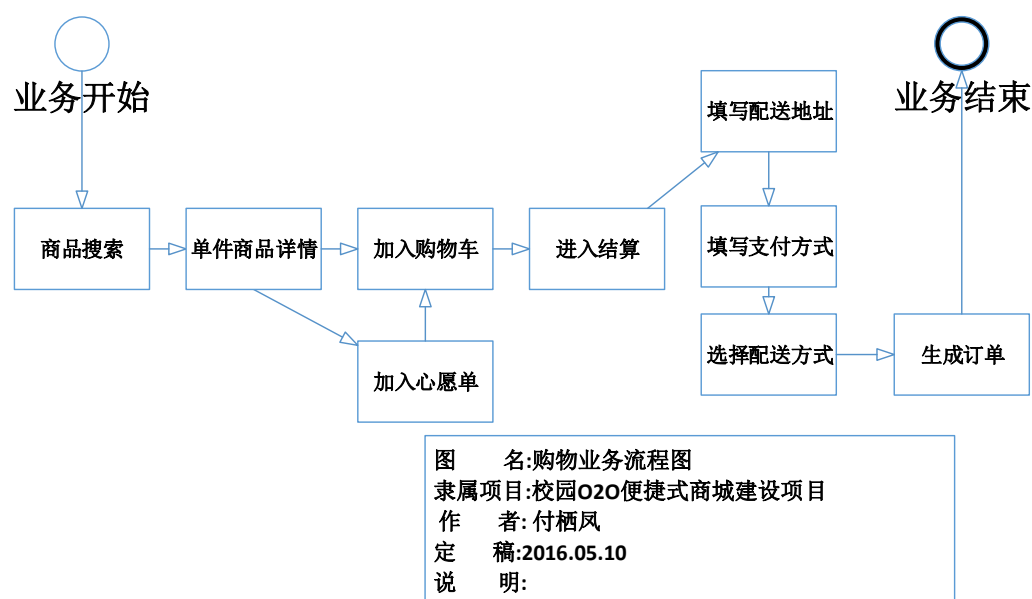


图 4.1 购物业务流程图

Fig.4.1 shopping business flow

4.1.2 用例建模

概念建模主要用于界定系统边界，从顶层进行系统架构设计。本项目主要利用 Enterprise Architecture 进行系统设计，

在业务分析过程中，最初以 amazon 购物网站购物流程得到本项目的概念业务流程，并通过调研分析得到系统的购物业务流程，如图 4.1 所示，基于它进行 Eriksson 业务建模、完成概念世界向信息世界的转变。转变过程为用例开发奠定了基础。表 4.1 列出了本文用例开发的用例清单，图 4.2、4.3 是用例建模后得到的，10 个用例的确立的意义在于：为软件工期的估算提供了一个重要的参考，通过它们能大概地衡量出系统的规模、并且为数据库架构的实现提供支撑。

图 5.1 定义并描述了“校园 O2O 便捷式商城建设”项目的软件架构。Webapp 组件包抽象了网站建过程中所有代码工作量；src 组件包抽象了所有 java 类及资源配置文件；而 config、test 组件包与 src 在代码工程中属同级资源包，config 中涉及的文件有 log4j.properties, applicationContext.xml, struts.xml，甚至还可以包含 cache.xml；test 中的包结构与 src 中的大致相同，主要内容定义是一些用于单元测试的测试类。

在本文项目中，所有的 Entity 类、DAO 类、Service 类、Action 类、sessionFactory、事务、类的注解、缓存、连接池、等资源都将配置在 applicationContext 当中。

表 4.1 购物流程清单

Table4.1 shopping UC list

用例编号	主用例名	命名	子用例名	子用例命名	用例场景举例
UC0101	商品搜索	SearchAction	关键词搜索	execute	主页输入关键词
UC0102			按类搜索	kind	单击下拉种类选项
UC0103			条件搜索	union	价格和种类联合筛选
UC0201	购物车操作	CartAction	商品添加	add	单击关联的添加按钮
UC0202			商品移除	remove	单击关联的删除按钮
UC0203			列项显示	execute	单击商品添加后显示
UC0301	心愿单操作	WishAction	商品添加	add	单击关联的添加按钮
UC0302			商品移除	remove	单击关联的删除按钮
UC0303			列项显示	execute	单击商品添加后显示
UC0401	配送地址操作	AddressAction	地址创建	execute	填写表单后提交
UC0402			地址修改	edit	单击编辑
UC0403			地址删除	remove	单击删除
UC0404			地址查看	look	单击查看
UC0501	支付方式操作	PayAction	支付新增	execute	填写表单后提交
UC0502			支付删除	remove	单击编辑
UC0503			支付修改	edit	单击删除
UC0504			支付查看	look	单击查看
UC0601	订单操作	OrderAction	订单生成	execute	填写表单后提交
UC0602			订单修改	edit	单击编辑
UC0603			订单删除	remove	单击删除
UC0604			订单查看	look	单击查看

续表 购物流程清单

UC0701	登录	LoginAction	登录	execute	顾客单击登录
UC0702			登出	logout	关闭浏览器
UC0801	信息修改	ModifyAction	信息修改	execute	单击修改密码
UC0802				exit	供应商单击退出系统
UC0901	注册	RegAction	注册	execute	顾客注册
UC0902			供应商注册	enroll	供应商注册
UC0903			注销用户	destroy	顾客、供应商注销
UC1001	商品操作	WaresAction	信息创建	create	新增商品项
			信息修改	edit	更改商品信息
			信息删除	delete	单击删除
			列项显示	execute	单击显示

表 4.2 账户操作及后台代表性用例清单

Table4.2 UC about account and representative MIS

UC0701	登录	LoginAction	登录	execute	顾客单击登录
UC0702			登出	logout	关闭浏览器
UC0801	信息修改	ModifyAction	信息修改	execute	单击修改密码
UC0802				exit	供应商单击退出系统
UC0901	注册	RegAction	注册	execute	顾客注册
UC0902			供应商注册	enroll	供应商注册
UC0903			注销用户	destroy	顾客、供应商注销
UC1001	商品操作	WaresAction	信息创建	create	新增商品项
			信息修改	edit	更改商品信息
			信息删除	delete	单击删除
			列项显示	execute	单击显示

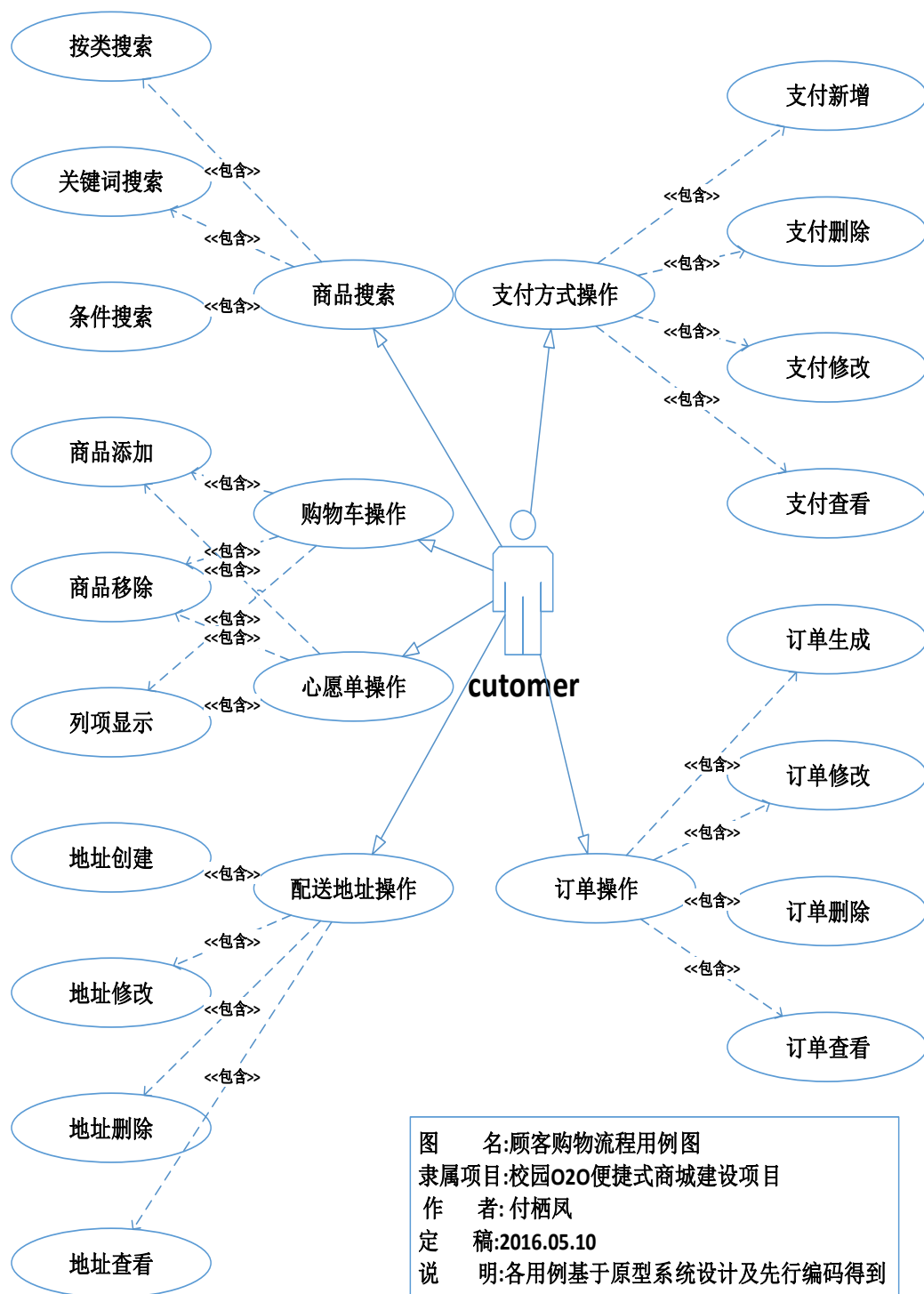


图 4.2 顾客购物流程用例图

Fig. 4.2 UCD about customer shopping

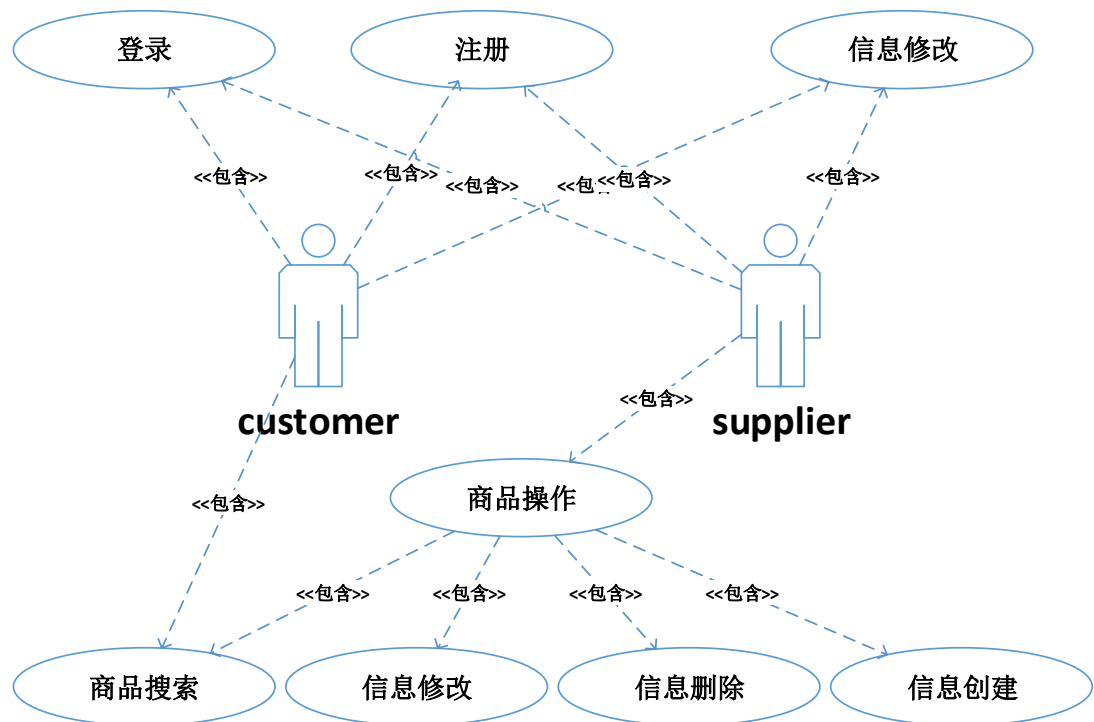


图 4.3 后台管理系统用例图

Fig.4.3 UCD about account and MIS

4.2 系统模块的设计

4.2.1 WebApp 设计

本文 Webapp 是指在 myeclipse 的 Web 项目中 WebContent 或 WebRoot 文件夹下所有文件。在 MyPro 项目, 主要采用 jsp 进行 MyPro 项目的 index.jsp、error.jsp、exception.jsp、my.jsp、my-edit.jsp、my-list.jsp 编码。目标系统 ShowPro 前台 jsp 置于 jsps/eshop 文件夹当中, 而后台则置于 jsps/mis 文件夹当中, 所有对象的 edit.jsp 置于 mis/edits 文件夹下, 所有的 list 置于 mis/list 下。

Shop 是本文所选择的拟进行代码复用的一个同样拥有 SSH2 软件架构的系统, 也是市面上常见的 SSH2 系统, 本文件的 MyPro 项目对它提出了两点改进, 一个是表单代码的数据绑定, 一个是 add 页面与 edit 页面进行合并, 另一个是表单提交的 action 配置, 因为采用了 struts2 约定, MyPro 中所有 action 类放到了 dashow.struts2 包下, 所以系统运行时会自动查找带有 Action 后缀的类, 并转换成.action, 第三个是对结果视图的配置采用注解方式进行。

4.2.2 数据库设计

本文所设计的数据库名为 `ecommerce`，数据库平台为 `mysql`，采用 `intelliJ idea` 进行 `sql` 语句操作。基于原型系统，加上用例开发的辅助，本文完成了数据库从概念级转向定义级，即从 `Chen's` 数据库表示法表示的 `ER` 图——暂未进行实体间的关系确定，转变成了 `IDEF1X` 表示法表示的物理数据——它还确立了数据库表与表之间的关系。从属关系在现实世界是树状结构，存储模型可以子类别主键作为父类别外键，有几级类别就建几个基本表，也可以一张商品信息表与一张类别表，类别表外加一个字段，将类别表主键作为那个字段的外键，从而简化表的建立，本文项目的数据库商品类别就设计成这样，`ecommerce` 数据库包含地址表（`address`）、购物车表（`address`）、顾客表（`customer`）、类别表（`kind`）、测试表（`mytable`）、订单表（`order`）、供应商表（`supplier`）、商品表（`wares`），主键采用代理主键，主键在系统开发过程中采用 `native` 主键生成策略。

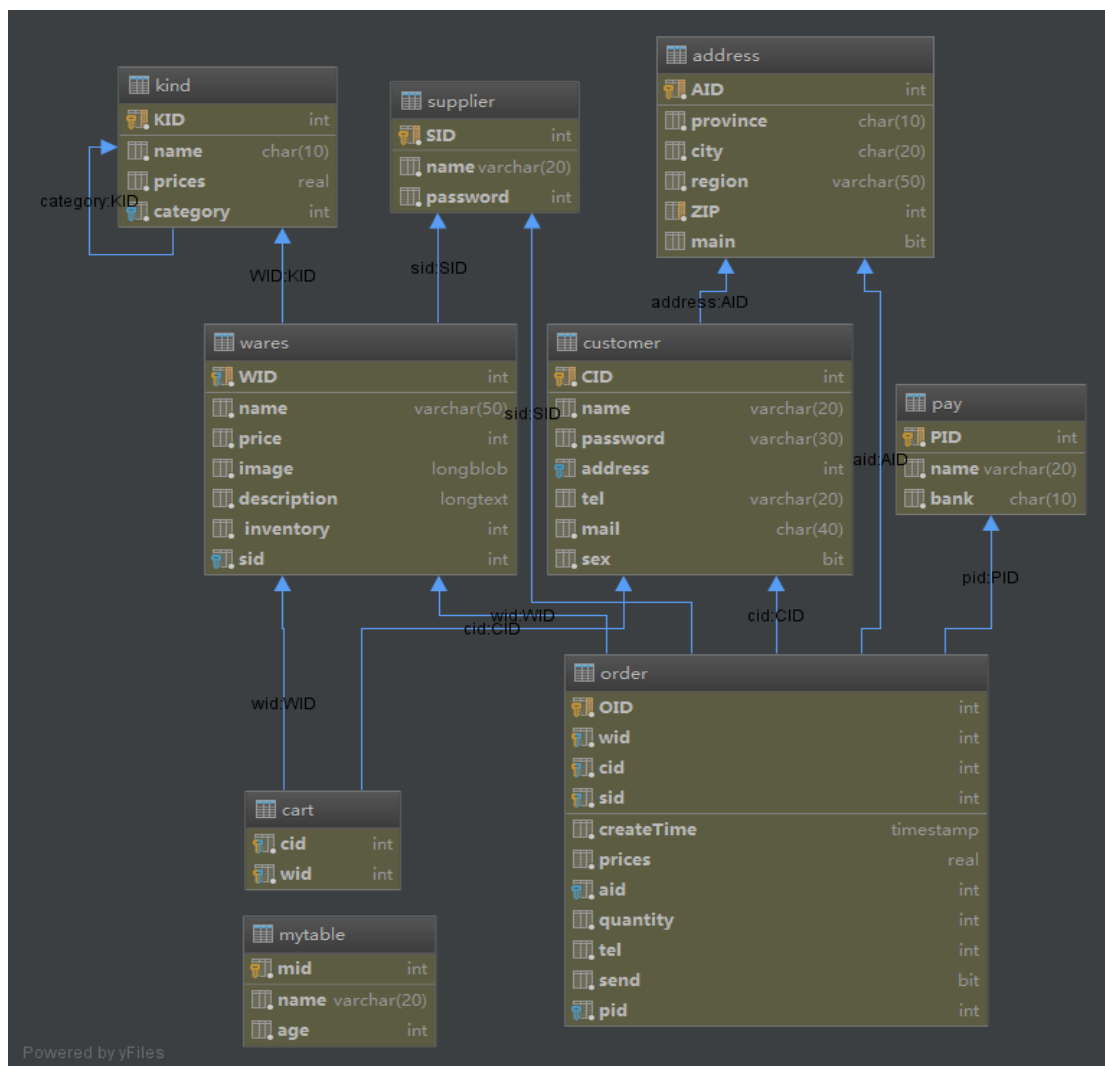


图 4.4 IDEF1X 表示法表示数据库物理架构图

Fig4.4 IDEF1X show physical database schema

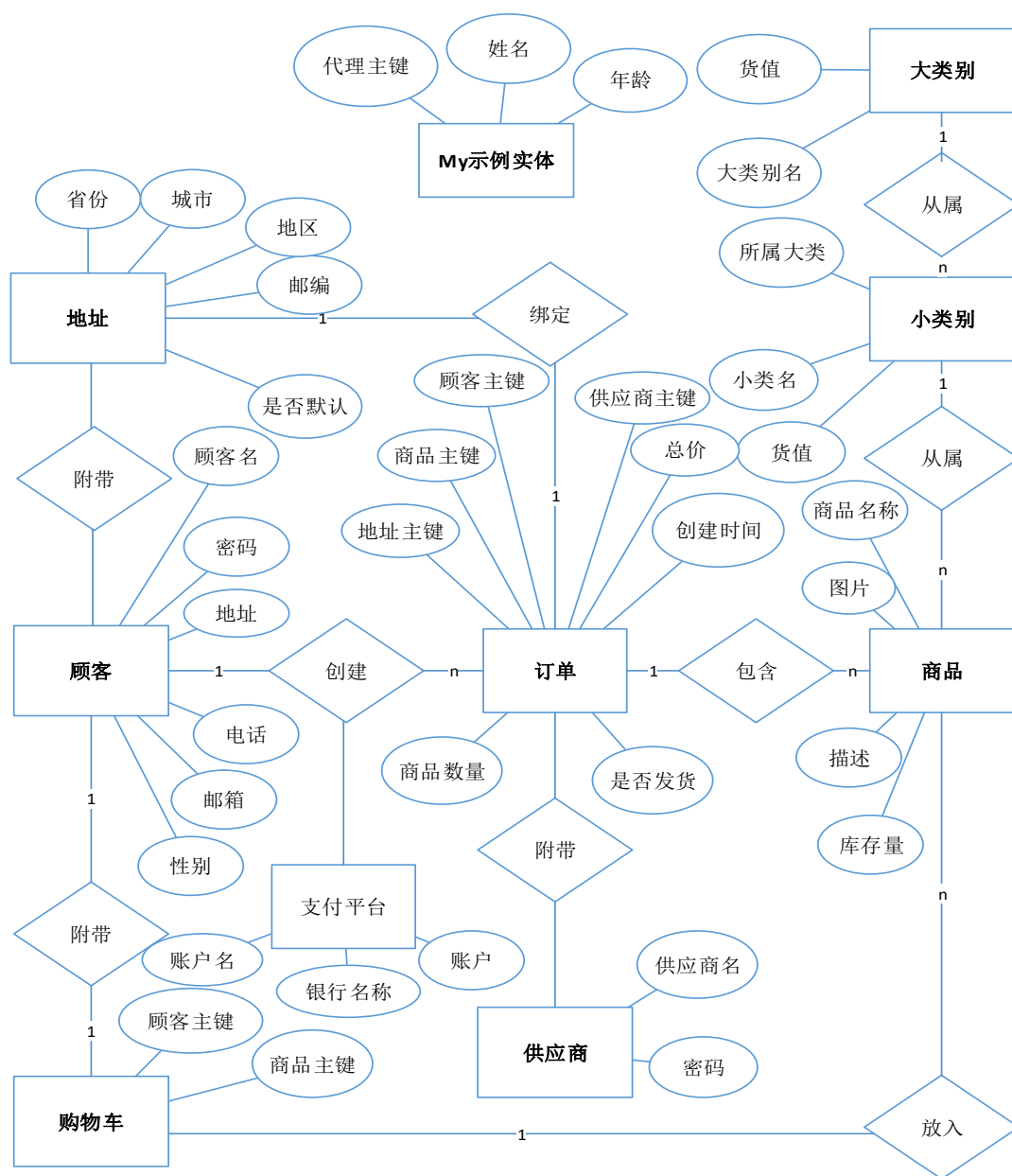


图 名:数据库概念实体关系图
表示方法:Chen's数据库表示法
隶属项目:校园O2O便捷式商城建设
作者:付栖凤
定稿:2016.12.10
说明:逻辑实体主键使用代理主键,属性从略

图 4.5 数据库概念实体关系图

Fig.4.5 DB of concept ERD

表 4.3 地址表

Table 4.3 address table

地址表 (address)					
列名	对应属性名	字段类型	主 键 约 束, null	备注	默认值
AID	代理主键	int	是, 否	自增	默认值
province	省	char(10)	否, 否		
city	市	char(20)	否, 否		
region	地区	varchar(50)	否, 否		
zip	邮编	int	否, 否		
main	默 认 配 送 地 址		否, 是		

表 4.4 购物车

Table 4.4 cart table

购物车信息表 (cart)					
cid	顾客主键	int	是, 否	外键	
wid	商品主键	int	是, 否	外键	

表 4.5 顾客信息表

Table4.5 customer table

顾客信息表 (customer)					
CID	代理主键	int	是, 否	自增	
name	顾客名	varchar(20)	否, 否		
password	密码	varchar(30)	否, 否		
address	地址	int	是, 是		null
tel	电话	varchar(20)	否, 是		null
mail	邮箱	Char(40)	否, 否		
sex	性别	bit	否, 否		0

表 4.6 类别表

Table4.6 kind table

类别表(kind)					
KID	代理主键	int	是, 否	自增	
name	小类别名	char(10)	否, 否		
prices	货值	float	否, 否		
category	所属大类名	int	是, 否	外键, 参 考 kind	

表 4.7 测试表

Table 4.7 mytable table

测试表(Mytable)					
MID	代理主键	int	是, 否	自增	
name	姓名	varchar(20)	否, 否		
age	年龄	int	否, 否		

表 4.8 支付表:

Table 4.8 pay table

支付表(pay)					
PID	银行帐号	Int	是, 否		
name	账户名	Varchar(20)	否, 否		
bank	银行注册名	Char(10)	否, 否		

表 4.9 订单表

Table 4.9 order table

订单表(order)					
OID	代理主键	int	是, 否	自增	
wid	商品主键	int	是, 否	外键	
cid	顾客主键	int	是, 否	外键	
sid	供应商主键	int	是, 否	外键	
aid	地址主键	int	是, 否	外键	
pid	支付方式主 键	int	是, 否	外键	
prices	订单价格	float	否, 否		0
createTime	时间戳	timestamp	否, 否		系 统 当 前 时 间

quantity	商品数量	int	否, 否		1
send	是否发货	bit	否, 否		0

表 4.10 商品信息表

Table 4.10 wares table

商品信息表(wares)					
WID	代理主键	int	是	自增	
name	名称	varchar(50)	否		
price	单价	int	否		
image	图片	longblob	否		
description	描述	longtext	否		
inventory	库存量	int	否		
sid	供应商主键	int	是		

表 4.11 供应商表

Table 4.11 supplier table

供应商表(supplier)					
SID	代理主键	int	是	自增	
name	名称	varchar(20)	是		
password	密码	int	否		

4.2.3 Struts2 模块设计

Struts2 模块设计主要涉及 Action 命名与设计、结果集的视图配置、拦截器设置、类型转换、action 命名空间等工作。重定向、增删改查等考虑。在本文所涉及的 MyPro 项目中, form 表单的 action 属性为“my”, struts2 框架运行时将自动查找名为 struts2 下的 action、actions 包, 并进一步查找是否存在 MyAction, 因此在本项目中, 利用这一条约定, 在后台代码开发过程中, 所有页面事件处理的类放在了 dashow.struts2.action 包中, 并创建了一个 MyAction 类, 这样便省去了在 struts.xml 配置文件中配置 action 的工作, 同时对创建的 MyAction 加上类级别的 Spring 注解@Controller 与@Scope("prototype")便将 struts2 的类交给 spring 管理。若要访问 MyAction 的方法 loading 方法, 在 jsp 页面中则可以通过 my!loading 进行访问, 其采用的技术为动态方法调用(DMI), 本质是 DMI 扫描 action 名字的“!”字符, 如果找到则“欺骗”框架来调用“!”后面的方法。

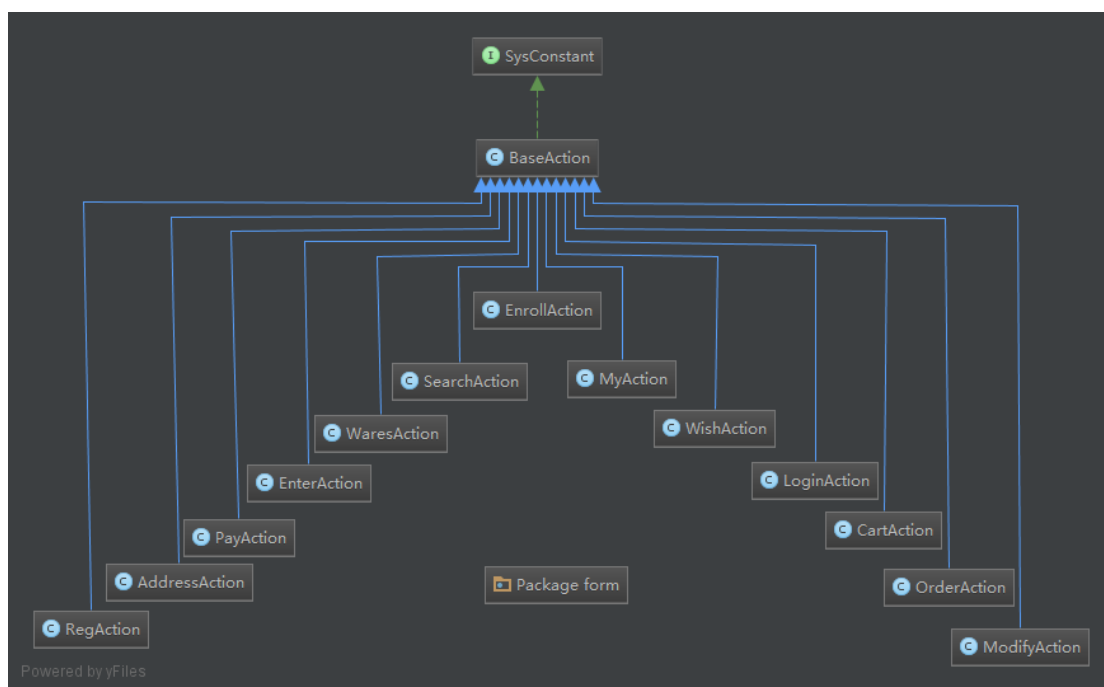


图 4.6 ShowPro 的 Struts2 层

Fig4.6 ShowPro' Struts2 tier

4.2.4 Spring4 模块设计

本层设计的工作主要是给 struts2 中的 Action 类和 operation 包的 DAO 类进行注解添加, 以及配置 applicationContext.xml 文件, 本层的存在主要将其它两层交与 Spring 管理, 同时, spring 的接口设计目的主要为后期引入 Tuscany 进行系统的分布式开发预留空间, 它将为系统后期的功能拓展与维护提供良好的便利, 在实际生产过程中, 有些规模较小的、不需要进行分布式的系统可以将本层砍掉。

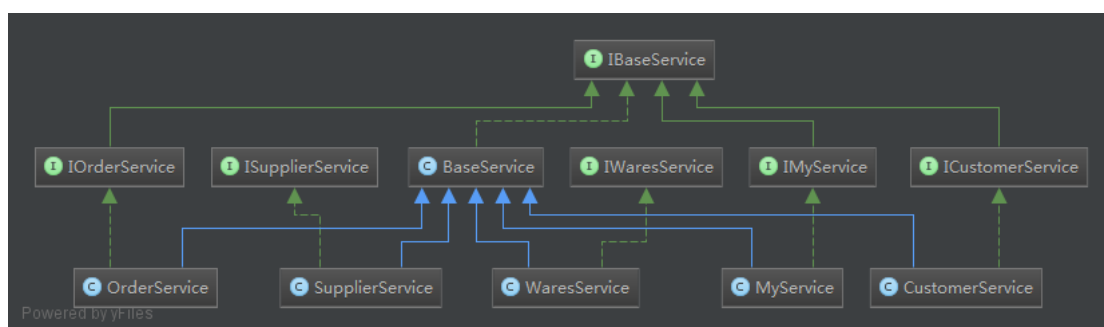


图 4.7 ShowPro 的 Service 层

Fig4.7 ShowPro's Service tier

4.2.5 Hibernate 的 DAO 模块设计

本层代码可以使用 myeclipse2015 的 Hibernate 逆向工程技术 (hibernate revers engineer)生成实体类时连带生成本层代码并略加修改、当系统完工后, 可

通过抽取接口，引用反射机制等进行代码优化。本层代码开发主要涉及少量的 hql 查询语句的编写。本项目的 MytableDAO 重点设计了基于元数据的查询方法。

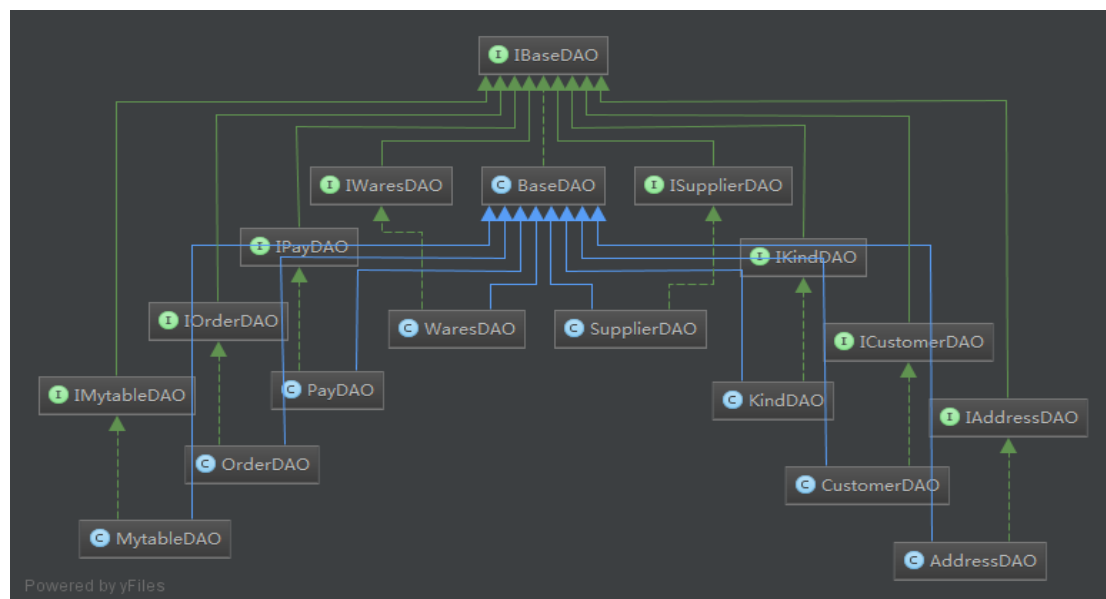


图 4.8 ShowPro 的 DAO 层

Fig4.8 ShowPro's DAO tier

4.2.6 Hibernate 的 Model 设计

当数据库设计完成后,在 Myeclipse2015 中新建完 Web 项目并添加 struts2.1、Spring3.1、Hibernate4.1 框架后,可采用 Hibernate 逆向工程技术自动生成注解配置的实体类,并将注解的实体、sessionFactory 及数据源、数据连接池等全部交给 Spring 管理。在实际应用是一般可以设置一个 BaseEntity 类,字段包含主键、删除位、时间戳、版本等。

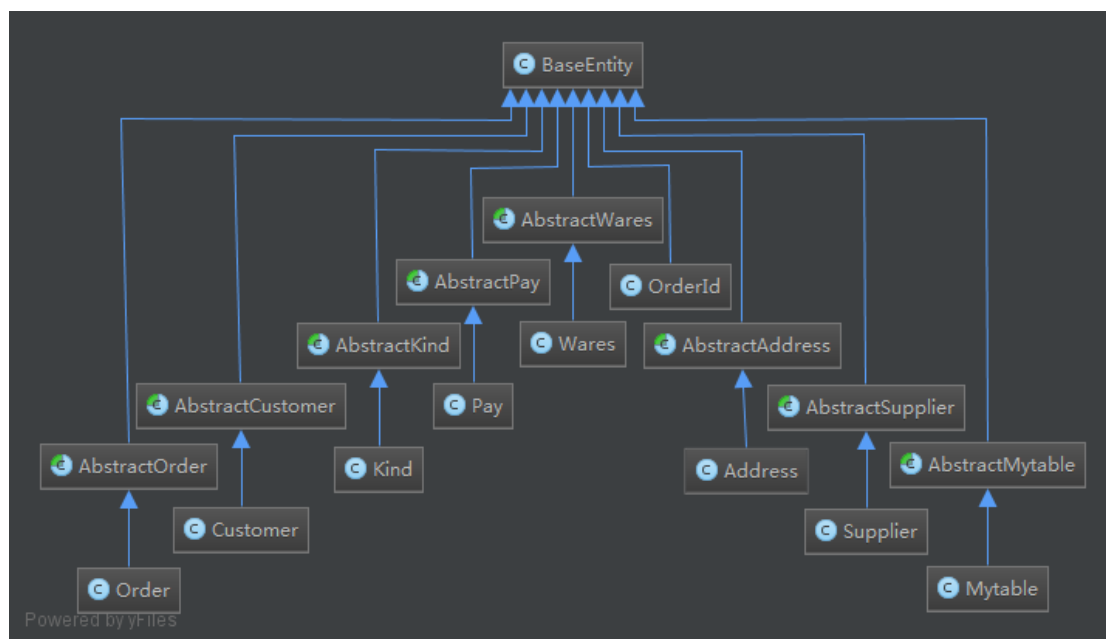


图 4.9 ShowPro 的 Model 层

Fig4.9ShowPro's Model tier

4.3 本章小结

在系统分析与设计阶段，通过不断的系统试验性编程，从而得到趋于稳必的软件架构，并对宏观的技术路线进行细节方面的调整，本文在此阶段采用的系统实验平台涉及 Tomcat8、JDK8、maven（用于版本控制），以及 SSH2 的 Jar 包的版本，Struts 与 Struts2 是两个有着正面天壤之别的框架，前期以 Struts 进行试验性编程，最终被迫返工，改用 struts2，以及由 tomcat8 和 jdk8 改回 tomcat7 和 jdk7 便是忽略了技术老化与开发工具的版本暂时不能很好支持 tomcat8 等因素。

系统分析与设计阶段核心产物是 6 个任务模块的确定、软件架构有了清晰的描述、数据库基本表及字段得以确定、项目规划阶段的培训任务基本结束。

第5章 系统编码与实现

5.1 系统任务分解

图 5.1 定义了目标系统开发所涉及的任务模块，采用 EA 的 组件模型（Component model）进行建模。Web model 与 Class model 对代码级文件进行了抽象，揭示了文件核心数据、变量、变量类型、方法、文件关系。而组件模型将它们推向了更高一级的抽象，组件包则将组件再次抽象。越高一级的抽象，越趋于走向一个通用的架构，SSH2 架构便是广泛运用的轻量级企业架构。合理的架构设计将决定系统的质量。

目标系统开发除了表 5.1 所定义的业务需求功能外，还涉及开发环境测试任务，中主要针对 Tomcat、JDK、JavaEE、web.xml、Struts2、Spring、hibernate、mysql、easyui、jslt 等的版本进行选择，并考虑诸如文件上传、图片存储等非功能需求。

在开发过程中，基于表 5.1 得到的结论是 Myeclipse2015 及 IntelliJ idea 对“以 Tomcat8+JDK8 进行 SSH2 系统开发”支持不是很好，JavaEE7 需要 JDK8 和 Tomcat8，才能接入 html5 应用。系统开发除了解到 10 个用户需求用例外，还应当涉及诸如图片加载。

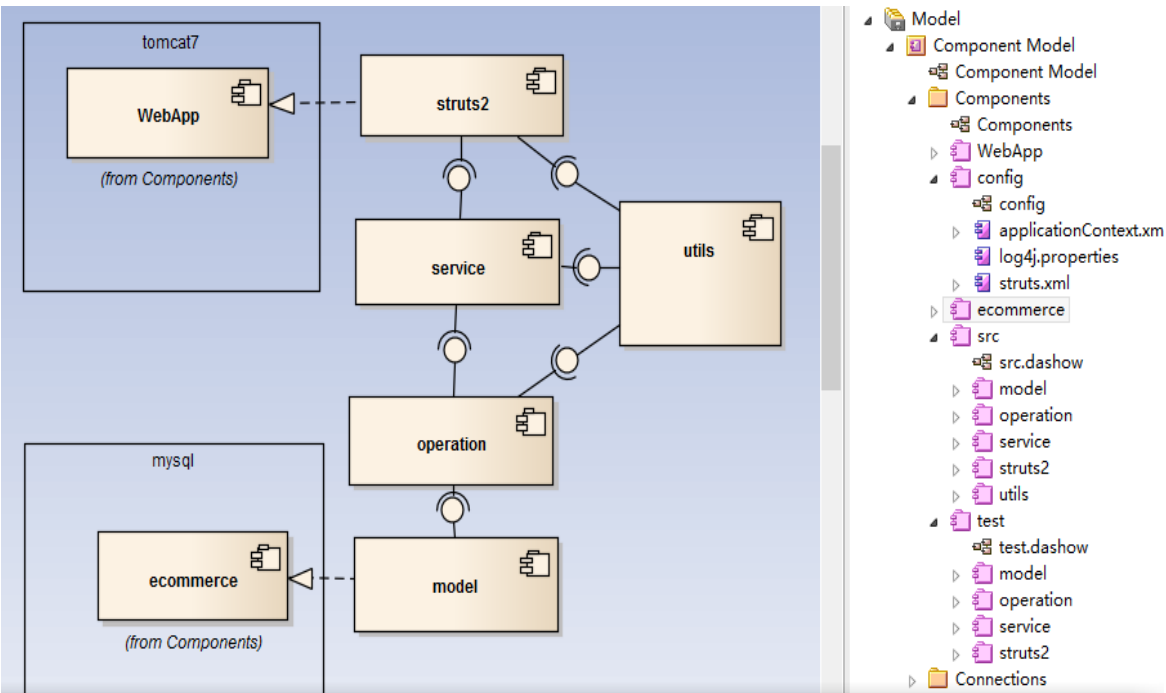


图 5.1 构建系统的任务模块——软件架构描述图

Fig5.1 task groups about integrating the system ——description software architecture

表 5.1 业务需求功能开发任务定义表

Table 5.1 task of business requirment function define

系统开发任务定义表 1								
用例编号	测试用例	用例名（类名）	webapp	Struts2	Spring	Dao	Hibernate	数据库
UC00	环境测试	MyAction						
UC01	商品搜索	SearchAction						
UC02	购物车操作	CartAction						
UC03	心愿单操作	WishAction						
UC04	配送地址操作	AddressAction						
UC05	支付方式操作	PayAction						
UC06	订单操作	OrderAction						
UC07	登录	LoginAction						
UC08	信息修改	ModifyAction						
UC09	注册	RegAction						
UC010	商品操作	WaresAction						

5.2 项目估算

软件项目估算对于软件项目管理、资源合理分配、风险管控有着重要的意义，它对于软件项目的成败举足轻重，国内对于软件项目估算的应用不是十分成熟，在国外，软件项目估算的成熟还推动着软件工程经济学的发展。项目规模估算有两种基本方法，本文基于以上的项目任务分解，进行基于 SSH2 项目估算活动。

(1) 估算完成项目规划阶段所需要的时间(Planning Phase Approach)

最简单的方式就是花大量的时间在“项目规划”阶段来预测整个项目的时间所需。对于本项目的项目规划过程，本人 2015.9.1 到 2015.10.31 期间，进行本项目的原型开发、数据建模、技术调研、文档拟制等规划活动，参考下表典型的关于 SDLC 的工业标准百分比，得出本项目所需时间为：

$$T_1 = 2 / 0.15 = 13.3(\text{人月}) \tag{5.2.1}$$

项目开启从 2015.9.1 起的话，因为

$$T_2 = 2 / 0.15 \times (0.15 + 0.20) = 4.67(\text{人月}) \tag{5.2.2}$$

即系统分析的完成理论上应当在 2016.1.20 左右完成。与本项目施工进展相吻合。

表 5.2 SDLC 的工业标准百分比表(U.S)

Table 5.2 Industry standard of SDLC phase percentage (U.S)

SDLC 的工业标准百分比				
阶段	规划	分析	设计	部署
百分比	15%	20%	35%	30%

(2) 功能点估算(Function Point Approach)。

功能点估算将子模块所需的功能将其转换成代码行，从而估算得到项目规模，一但确定其规模，便能估算出完成项目所需要的时间，从而进行统筹优化，制定合理开发进度，分配相关的人力物力，本文功能点估算的前提前置条件是一个用例开发完成，因为它的代码风格决定了其它用例的风格，在估算过程前，jsp 页面多少的估算可以通过 web 的建模来记录，MyPro 项目的 my 对象，共有 16 个功能点，即：

$$\text{Edit.jsp} + \text{List.jsp} + (\text{Struts2} + \text{Spring} + \text{DAO}) * 4 + \text{Hibernate} + \text{DataBase} = 16 \tag{5.2.3}$$

4 代表增删改查 4 个功能点。16 个功能点主要针对 java 代码。

在全局范围内，index.jsp（主要用于收集客户端信息并处理、然后再跳转到其它界面）、error.jsp、exception.jsp、home.jsp（前台商品展示主界面）、admin.jsp（后台管理系统主界面）、各用例的 edit.jsp 和 list.jsp，各 jsp 都需要编写一定的 html 代码，加上用于顾客服务及商品展示的 9 个 html 原型系统界面。共计 34 个静态 html 代码开发。

Java 功能点转换成代码行的系数为 55，而 html 为 15，所以得出本系统的代码行 L 及工期 M 为：

$$L = 10 \times 16 \times 55 + (10 \times 2 + 5 + 9) \times 15 = 9175(\text{行}) \tag{5.2.4}$$

$$M = 1.4 \times (L/1000) = 13.03(\text{人月}) \tag{5.2.5}$$

两种估算方法相差 9 天。

5.3 软件开发策略

本文制定的软件开发策略分三步走：

(1) 第一步是基于 UC00 用例的 MyPro 子项目开发。

(2) 第二步是基于 Shop 子项目的二次开发。

(3) 目标系统 ShowPro 功能开发。

(4) ShowPro 功能布局。

MyPro 子项目设立原因、目标及意义：MyPro 项目作为“校园 O2O 便捷式商城”项目的子项目，其作用是调整项目规划阶段的工作，剔除不合理的设定；总结前期先行建模、测试编码等工作；对当前同型 SSH2 系统弊端或者不足进行实验并提出改进方案；建立目标系统 ShoPro 项目开发过程中涉及的标准，标准内容有文档标准、代码风格标准、制图标准、过程标准等；深化软件架构及代码工程结构。增删改查（CRUD）是对象的基本操作，基于 Mytable 基本表进行 CRUD 功能开发流程，有利于加快后期开发工作。本文对 MyPro 项目的描述，也是对“校园 O2O 便捷式商城”目标系统 ShowPro 的描述，因为前者是后者的理论指导。

Shop 子项目设立原因、目标及意义：第一，本文设计的数据库相对复杂，没有进行过多验证，后期还有一定的工作量需要做，shop 可以完成对它的验证工作；第二，前端美工从零开始进行精细化设计从技术上存在一定困难，同时也没有必要进行，完全可以采用代码复用的方式进行，Shop 提供了代码复用的可能。第三，Shop 本身代表目前落后的系统，针对 MyPro 项目的完成，利用它的先进性，对它进行二次开发与改进，有利于使目标系统 ShowPro 推向软件主流市场。

5.4 子项目 MyPro

5.4.1 MyPro 的系统分析与设计

图 5.2 是本文的 Eriksson-Penker Model 业务建模。业务建模尝试捕获到业务需求，“Shopping”又是核心关注点，本文为其设立一个集成线包“experssion tier”，集成线包对软件系统进行了抽象，里面有若干读写对象，从业务过程到读写对象的连线实质上是一系列的用例，并引发下一步用例建模(use case model)。

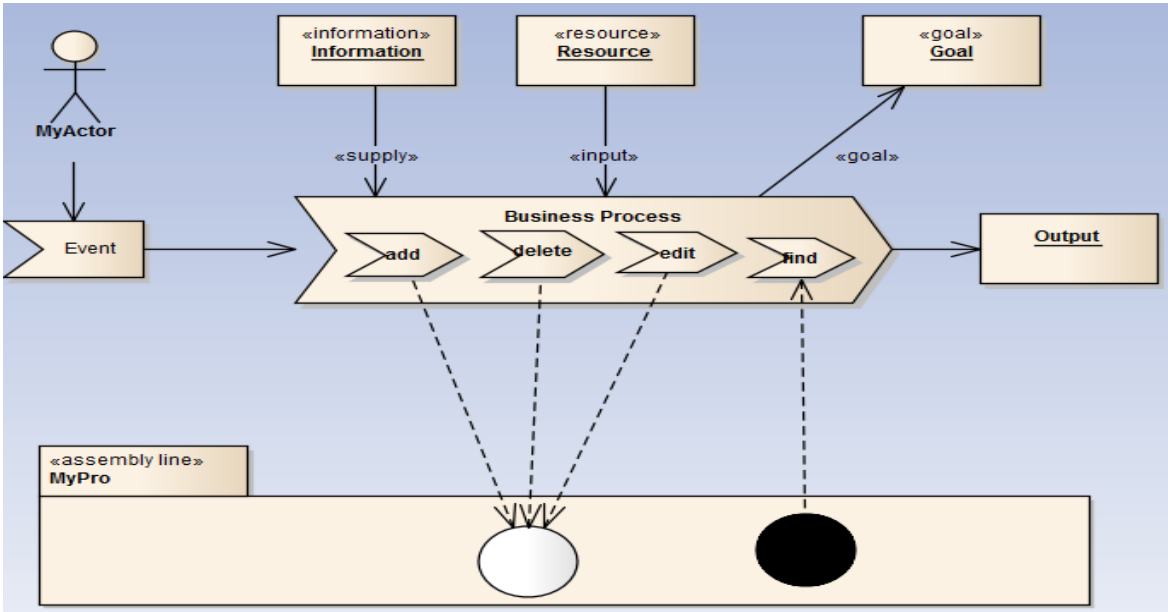


图 5.2 项目的业务建模

Fig5.2 Business model about project

Use case model 用例模型。基于前一步，我们开始用例画图，首先确立相关参与人员，进一步确定相关用例，先期用例有些可能有用，有些是多余的，需要进一步的合并、确立、评级等工作，完成之后，我们再进行用例规约，规约可得到《用例规格说明书》，这份文档是我们最终所需要的，该文档传递着包含着描述系统信息流的数据模型，以及描述业务流程的进程模型。《用例规格说明书》中的内容将并入到《软件需求规格说明书》当中。本文基于用例建模过程的产物主要有三部分，一个是用例图，第二个是用例规约表、第三个是 IPO（input-process-output）图。IPO 是对用例的细化描述，对用例涉及到的数据进行记录，这些记录来自之前开发的原型，这些活动实质也是数据库设计的一个子活动，IPO 所需处理的数据的定义过程也是数据库实体及属性的确定过程。本文在此阶段，确立了 10 个主用例，并将基于用例的 IPO 图写入到《软件需求规格说明书》当中。

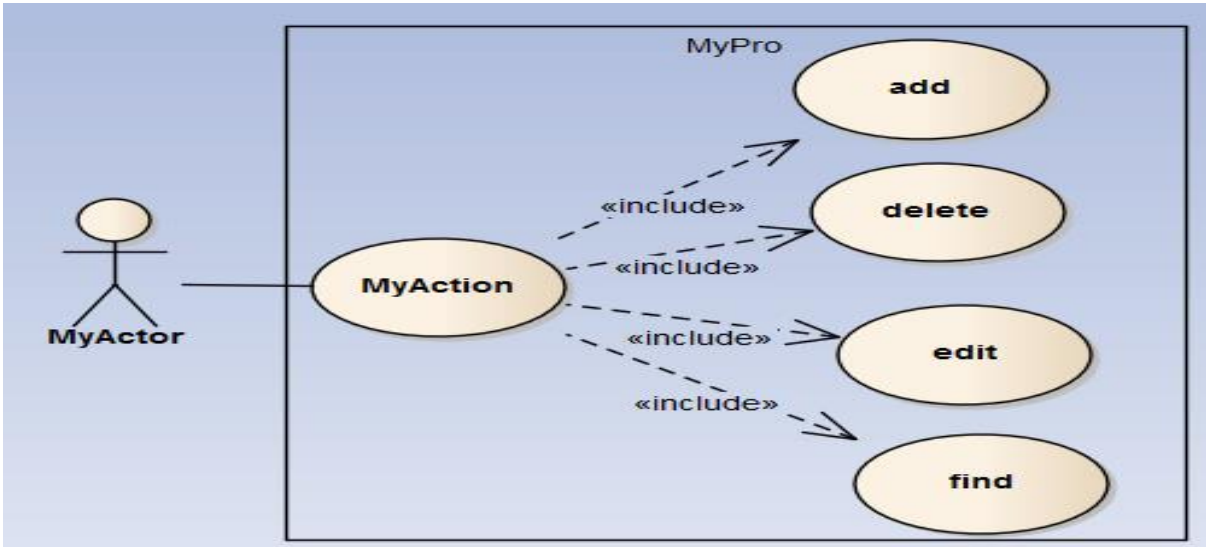


图 5.3 项目的用例建模图

Fig5.3 UCD

Web modeling Web 建模。通过前面几步关键活动，我们确定客户对系统的基本业务需求以及系统对数据处理的需求。原型系统的 html 文件与.java 类文件一样，终究是代码级的详细，无法显示它们的关联性，在软件设计、项目组成员交流时，必然有想看到它们关系的需求，这时，我们便能通过 web modeling 技术像建立类图一样，对 webapp 文件进行抽象。

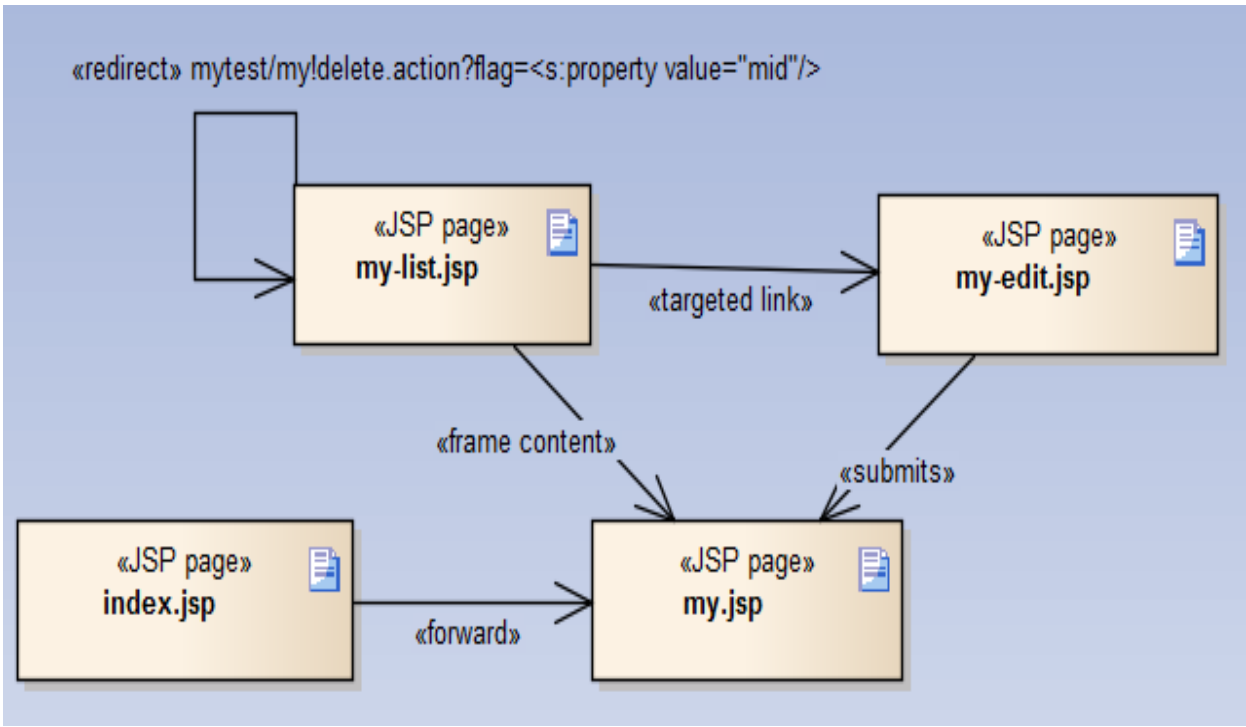


图 5.4 项目的 Web Modeling 图

Fig5.4 Web Modeling

项目的用例开发。用例命名 UC0102 中 UC 为 use case 缩写，01 代表主用例编号，02 为子用例标识。

表 5.3 用例表
Table5.3 UC table

用 例 编 号	主 用 例 名	命名	子用例名	子 用 例 命 名	用例场景举例
UC0001	My 用例	MyAction	新增	add	
UC0002			删除	delete	
UC0003			修改	edit	
UC0003			查询	find	

表 5.4 变形的 IPO 图

Table 5.4 distorted IPO table

输入		主用例：MyAction UC00		输出	
输 入 信 息	数 据 模 型	类型		输出信息	数据模型 类型
关键词	keyword	String		全部对象信 息	list List<mytab el>
mytable 对 象 表 单	myForm	mytable.mid mytable.name mytable.age			

5.5 用例规约表

Table 5.5 UC specification

用例名:MyAction My 用例		ID: U00	级别: 1
主要参考者:MyActor			
概述:参与者 MyActor 访问系统对 Mytable 实体进行 CRUD 操作。			
触发器:单击查询、新增、修改、删除等按钮进行操作。			
方式:手工输入表单、查询所用的关键词，其它则单击操作链接。			
前置条件: 无			
主要输入		主要输出	
描述	资源	描述	目标
关键词	String 类型	Mytable 对象集合	my.jsp
mytable 表单	mytable 对象	保存或更新数据库，再次查询	my.jsp
事件流		事件中的信息流	
1.运行系统 2.单击新增 3.填写表单 4.提交表单 5.单击“查询”进行关键词查询 6.在 list 对象集合中选定一个对象 7.单击修改 8.进行修改 9.在 list 对象集合中选定一个对象 10.单击删除。		1.系统运行后执行 my!list，查找所有对象并输出。 2.单击进入 my-edit.jsp，由单击新增所传参数，判断是新增还是修改，若是修改，加载并显示对象信息。 3.提交表单时依参数进行判断并操作进行库新增或更新。完成后执行 my!list 4.单击在删除，在数据库执行删除操作后，再自动转向执行 my!list 查询。	
约束: 未设置			

5.4.2 MyPro 的数据库设计

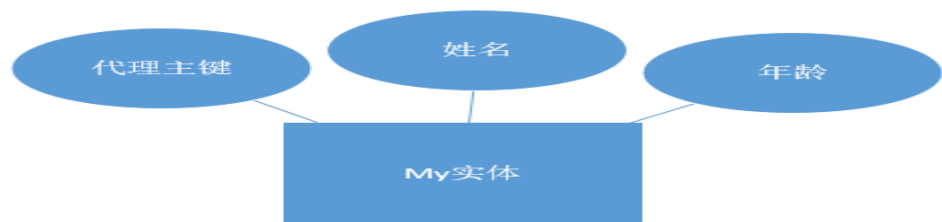


图 5.5 Chen’s 数据库表示法表示概念实体

Fig5.5 Chen;s ERD

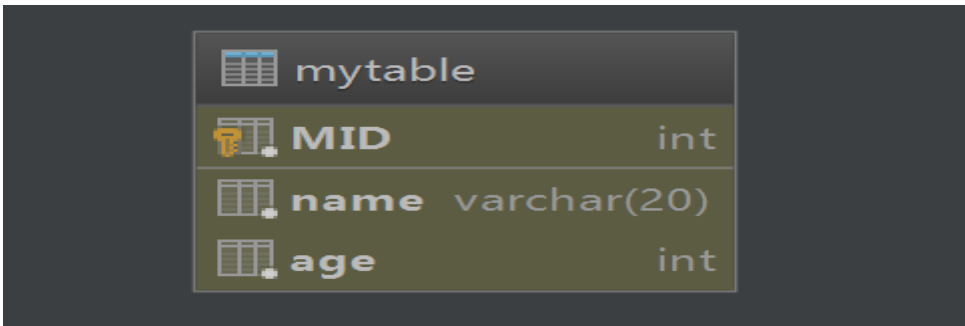


图 5.6 IDEF1X 表示数据库

Fig5.6 IDEF1X desription DB

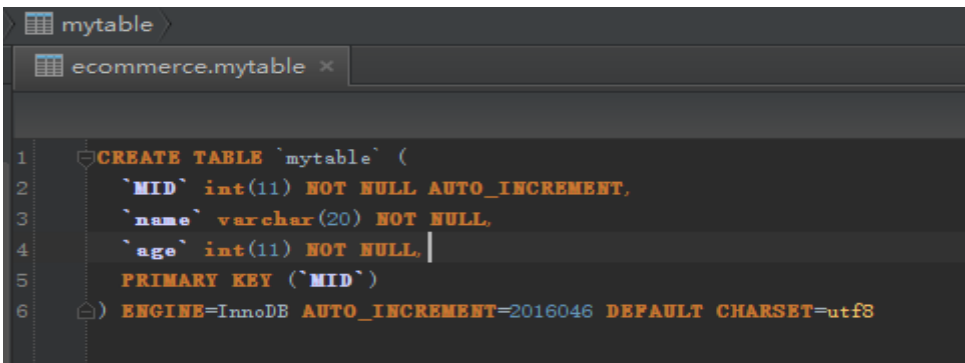


图 5.7 数据库代码

Fig5.7 DB SQL

表 5.6 mytable 测试表

table 5.6 mytable table for test

mytable 表(mid)					
列名	对应属性名	字段类型	主键约束, null	备注	默认值
MID	代理主键	int	是, 否	自增	
name	名称	varchar(20)	是, 否		
age	年龄	int	否, 否		

5.4.3 MyPro 系统编码与实现

Struts2 模块涉及到的包主要有 dashow.struts2.action 和 dashow.struts2.form 两个包。SysConstant 接口定义了 Action 中 CRUD 方法处理返回的字符串常量。BaseAction 实现了该接口并继承框架中的系统类 ActionSupport(所有 Action 都要直接或间接继承它)，在 BaseAction 中定义了基于 CRUD 的 loading、update、delete、list 四个方法，四个方法返回的字符串常为 SysConstant 接口中定义的，MyAction 类通过继承它，并采用注解的方式为这些字符串匹配了对应的结果集（即需跳转到的 jsp 或者需重定向到的 action）。同时采用注解完成命名空间配置、交给 Spring 管理等工作。MyAction 通过 IMyService 对象与 MyService 进行连接。

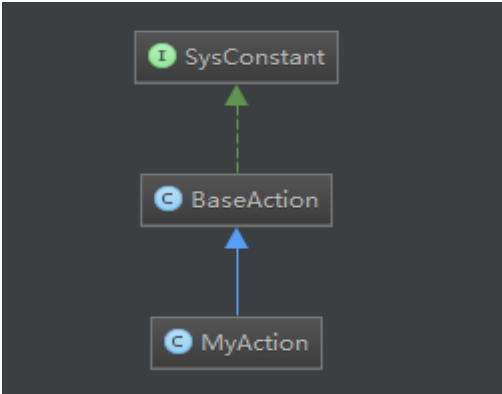


图 5.8 Struts 的 Action 模块描述

Fig5.8 description Action block of Struts2 tier

Service 主要用于连接 Struts 层与 Hibernate 层，在一些系统企业应用中，有的会选择砍掉它，因为它的基本是就是一个调用语句，中间没有多少业务功能处理，不过在系统进行分布式时，它的作用便会显现出来（将 IMyService 在客户端与服务器端分另申明为远程接口和本地接口）。

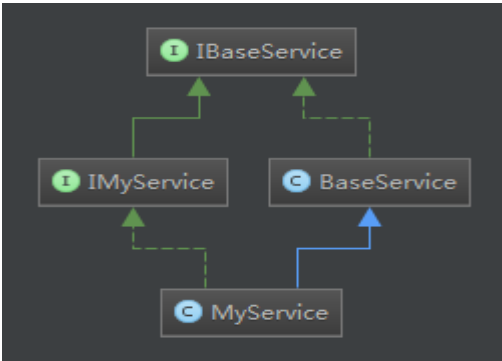


图 5.9 Spring 的 Service 模块描述

Fig5.9 description Service block of Spring tier

DAO 层通过 sessionFactory 创建 session 并进行数据库操作，采用事务注解方法申明事务，采用 hql 语句对实体类操作。

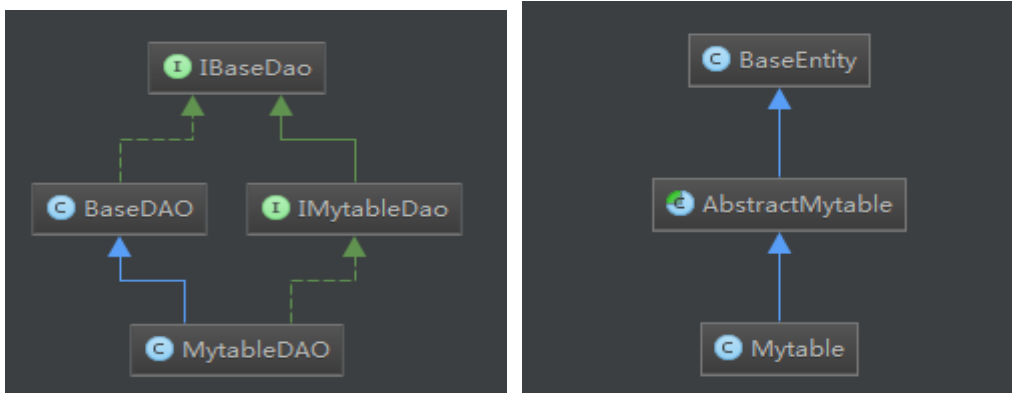


图 5.10 Hibernate 的 DAO 模块描述 图 5.11 Hibernate 的 Entity 模块描述

Fig5.10 Hibernate's DAO block Fig5.11 Hibernate's Model block

MyPro 系统运行后将直接跳转到 my.jsp，并通过运行 `<s:action name="my!list" executeResult="true" />` 此句，即调用 MyAction 的 list 方法，并将执行结果显示在 my.jsp 这一当前界面。单击 list 方法执行后显示的结果，单击新增信息或 修改信息，都将执行 my!loading，此方法通过判别所带参数 mid 是否为空，决定创建还是修改。若创建，my-edit.jsp 表单显示为空，否则显示主键为 mid 的信息，my-edit.jsp 表单单击保存执行 MyAction 的 update，同样根据 mid 判断是新增还是保存，保存完后，跳转到 my.jsp 并再次执行 `<s:action name="my!list" executeResult="true" />` 删除时则采用了重定向。只显示 my-list.jsp 内容。

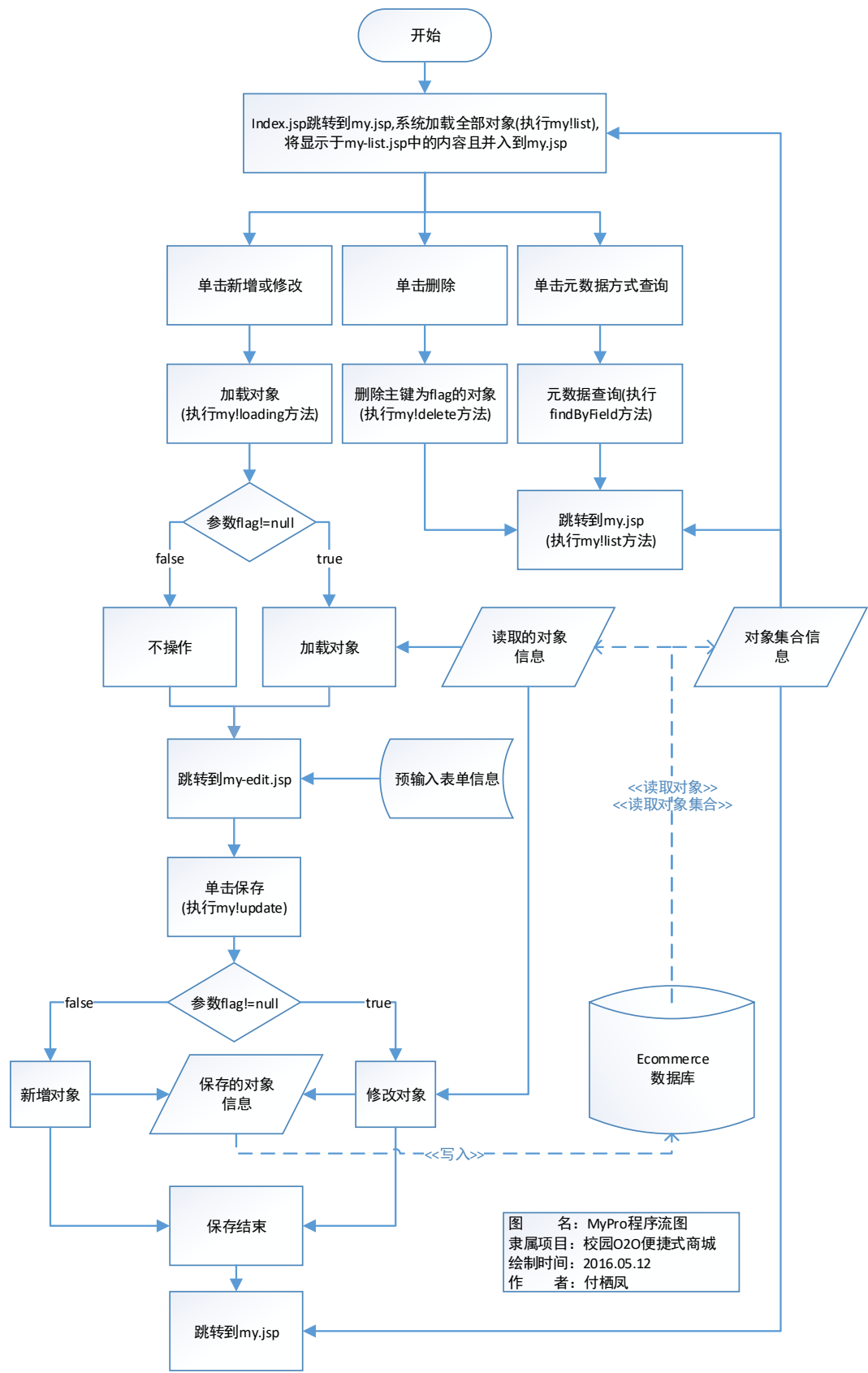


图 5.12 程序流程图

Fig5.12 Program flow of MyPro

5.5 子项目 Shop

子项目 shop 的软件架构是在 MyPro 的基础上通过砍掉了 Service 层,Strutsn 层通的 Action 实现接口 ModelDriven<T>中的 getModel 方法，直接操作领域模型（即实体类）。图 5.3 是 Shop 的主界面，通过主界面可以完成商品搜索、通过跳转到子界面，完成系统的登录、注册、查看商品详情，商品详情描述如图 5.4，单击商品购买的前提是登录了系统。单击购买后，跳转到购物车界面。



图 5.13 系统主页面

Fig5.13system’ home page



图 5.14 商品信息详情描述界面

Fig5.14 page of product detail info



图 5.15 购物车界面

Fig5.15 cart page

商家通过登录进入如下所示的后台界面，分别进行类别添加、查看所有类别、添加商品、查看所有商品、查看查询订单等操作



图 5.16 商品类别添加界面

Fig5.16 product add page

ID	类别名称	子类别	添加子类别	所属父类	编辑	删除
1	零售	有9个子类别	添加	无		
83	83号区	有8个子类别	添加	无		
126	126号区	有3个子类别	添加	无		

首页 上一页 [1] 下一页 尾页

图 5.17 查看所有商品类别界面

Fig5.17 page of list all product categories

ID	商品名称	所属类别	采购价格	销售价格	是否推荐	适合性别	编辑	删除
2	富龙卤蛋	其它	52.0	2.2	true	男		
3	润田矿泉水	其它	52.0	16.0	true	男		
4	马来西亚咪咪	其它	12.0	18.0	true	男		

首页 上一页 [1] 下一页 尾页

图 5.18 查看所有商品信息的界面

Fig5.18 list all product

后台导航栏

类别管理
添加类别
查看所有类别
商品管理
添加商品
查看所有商品
订单管理
查看订单
订单查询

添加商品

商品名称：

选择类别：零售

采购价格：

市场价格：

销售价格：

是否为推荐：☐是 ☒否

适应性别：男

上传图片： 未选择文件。

商品说明：

提交

图 5.19 进行商品添加的界面

Fig5.19 product add

编辑商品

商品名称：

选择类别：

采购价格：

市场价格：

销售价格：

是否为推荐：☒是 ☐否

适合性别：男

上传图片：

商品说明：
哼！喝润田，找一样可以脉动

提交

图 5.20 进行商品编辑的界面

Fig5.20 edit product

订单号	总金额	消费者	支付方式	创建时间	订单状态	修改
201004201352400389678	120.0	fooxe	邮局汇款	2016年05月11日 00:00	已发货	更新订单状态
201004220944580771720	120.0	fooxe	邮局汇款	2016年05月11日 00:00	已发货	更新订单状态
201004220957520484473	0.0	fooxe	邮局汇款	2016年05月11日 00:00	已发货	更新订单状态

首页 上一页 [1] 下一页 尾页

图 5.21 查看所有订单界面

Fig5.21check all order items

订单查询

订单号：

订单状态：

☒已发货 ☐交易完成 ☐已取消

会员名：

收款人：

提交

图 5.22 订单查询条件输入界面

Fig5.22 order query’s condition input page

5.6 本章小节

本章基于系统编码与实现阶段的工作，完成了项目开发任务的分解；完成了项目规模及工期的估算；完成了 MyPro 项目的一次开发，包括 CRUD 及元数据查询；完成了 ShowPro 的顾客和供应商的注册、登录、信息修改，完成了 Supplier、Address、Customer 等表的增删改查，为后期功能分配到各页面奠定了基础；进行了 shop 项目的二次开发主要定合理的代码工程结构有利于系统的后期维护。

第6章 文档编制

本文项目过程中的文档产物及内容概述如下：

(1) 《Dashow 标准》该文档主要对本项目所涉及文档进行风格统一，文档暂时只记录了来自理论综述类文献[02]的 DFD 片断说明模板、场景描述模板、测试计划模板、程序说明模板、需求收集模板等，这些模板为项目后期规范化提供参考。

(2) 《电子商务应用调研报告》该文档给出了电子商务应用平台、电子商务信息安全等方面的调研结果，从宏观构建起本文项目大概所处的社会环境，以便于降低开发人员对应用的不熟悉这一风险。

(3) 《数据库设计说明书》主要详细记录了 ecommerce 数据库以 Chen's 数据库表示法绘制的实体关系图、IDEF1X 法绘制的物理数据库、基本表的详细说明、sql 语句；以及暂未开发的触发器、存储过程、视图等。

(4) 《软件需求规格说明书》定义了前台业务流程、后台功能架构、用例图（UCD）、数据输入输出图（IPO）。

(5) 《软件设计说明书》主要定义了系统后期需要考虑开发的功能、资源配置文件的配置内容、待权衡的缓存策略(read-only, read-write, nonstrict-read-write, transactional)等。另附人机界面原型系统。

(6) 《SSH2 项目描述图》主要以图的形式记录了本文项目过程中产生的一系列内容，包括软件架构描述图、项目开发进程图、项目知识调用概要图。

(7) 《MyPro 项目报告》由于目标系统 ShowPro 在项目从需求分析到项目任务分解涉及 10 个用例开发，表及图纸较多，因此产生本报告，报告也是作为目标系统的理论运用、开发过程、技术水平、软件架构等的一系列体现。

(8) 《异常解决方案》记录系统开发过程中遇到的问题，从而加快后期的产生的同种异常的解决速度，也为后期 SSH2 系统研究提供材料。

(9) 其它文档，visio 文档描述诸如 ER 图、用例图、Gantt 图、Pert 图，系统 UML 建模文档 showqiu.eap，等等文档分别在系统构建过程中起重要作用。

结束语

(1) 本文的优点。

里氏开闭原则（OCP）是面向对象设计中“可复用设计”的基石，是面向对象设计中最重要原则之一，而本文所设计的软件架构，以“高内聚，低耦合”的模块化特点，在 MyPro 及 ShowPro 中充分地体现出。由于高度的模块化，使得系统在后期引入 Web Service，进行系统的分布式带来了极大的便利。通过在对 Spring 模块的 Service 的各接口上声明远程接口与本地接口，便能完成这一工作。

MyPro 的系统开发生周期（SDLC）为企业规范化软件生产有良好的借鉴作用。

(2) 本文不足之处

由于时间仓促，本文工作任务重心偏向于软件工程上游的项目规划、系统分析与设计上，而在系统编码与实现阶段，仅完成 MyPro 子项目，及 SowPro 的部分功能和 Shop 的局部二次开发局部。

(3) 展望

基于软件架构的理论研究，或者基于 Webservice 的分布式开发，MyPro 是一个很好的研究材料，而将理论成果转化为市场产品，借 MyPro 的先进性，ShowPro 将会显示很强的市场竞争力。

致谢

大学四年，从 2012 年 11 月 20 日到 2015 年 10 月 1 日，感谢信息学院能一直为我提供为机房守夜的工作，以及大二、大三两个暑假让我留在学校做实验。感谢我的导师对毕业设计提供的耐心支持；感谢大二领我入 java 世界还教我的软件工程的启蒙老师；感谢曾经呆在课题组时教导我的那位系统架构设计师，如明灯一样照亮我前进的路；还有默默支持我前进的其他老师，晚生无以为表，唯一一谨记。谨祝母校在你们的汗水浇灌下，涌现出更多的特色人才！

参考文献

- [01] 孙东川, 林福永. 系统工程引论. 北京: 清华大学出版社, 2004.
- [02] Alan Denis, Barbar Haley Wixom, Roberta M. Roth. System Analysis and Design 3th Edition. 北京: 人民邮电出版社, 2010.
- [03] 骆斌, 丁二玉. 需求工程——软件建模与分析. 第二版. 北京: 高等教育出版社, 2014. 12.
- [04] Hans-Erik Eriksson, Magnus Penker. Business Modeling with UML Business Patterns at Work. 机械工业出版社, 中信出版社, 2004. 3.
- [05] [美] Sharon Allen. Data Modeling for Everyone. 北京: 清华大学出版社, 2004. 9.
- [06] [美] Kurt Bittner, Ian Spence. Use Case Modeling. 北京: 清华大学出版社, 2003. 8. 1.
- [07] 冯冲, 江贺, 冯静芳. 软件体系结构理论与实践. 北京: 人民邮电出版社, 2004. 1.
- [08] 温 YI. 软件架构设计——程序员向架构师转型必备. 第 2 版. 北京: 电子工业出版社, 2013. 8.
- [09] [美] Barry W. Boehm. 软件成本估算 COCOMO II 模型方法. 机械工业出版社.
- [10] [美] Paul J. Perrone, Venkata S. R. “Krishna” R. Chagant, Tom Schwenk. Java EE 开发使用手册. 北京: 电子工业出版社.
- [11] 马洪江, 周相兵, 余堃等. 开源魅力——面向 Web 开源技术整合开发与实战应用. 北京: 清华大学出版社, 2013. 4.
- [12] 秦成德, 王汝林. 移动电子商务. 北京: 人民邮电出版社, 2009. 7.
- [13] 刘军, 董宝田. 电子商务系统的分析与设计. 高等教育出版社, 2003. 7.
- [14] 陈德人, 施敏华, 吴志航等. 电子商务系统结构. 高等教育出版社, 2002. 8.
- [15] 张利, 杨俊清. 电子商务设计师教程. 北京: 清华大学出版社, 2013. 8.
- [16] 求是科技. Visual Basic 开发 ERP 系统实例导航. 北京: 人民邮电出版社, 2005. 1.
- [17] [罗] Cristian Darie, [美] Karli Watson. ASP. Net 电子商务开发实战. 北京: 人民邮电出版社, 2010. 4.
- [18] 陈松洗. Java EE 电子商务系统开发从入门到精通——基于 Struts 和 Hibernate 技术实现. 北京: 清华大学出版社, 2007. 5.
- [19] 软件开发技术联盟. JavaWeb 自学视频教程. 北京: 清华大学出版社, 2014. 12. 第 20 章.
- [20] [美] David C. Hay. Requirements Analysis From Business views to Architecture. 北京: 清华大学出版社.
- [21] 邵维忠, 杨芙清. 面向对象的系统分析. 北京: 清华大学出版社, 2007.
- [22] 张晞. 以 UML 用例图为基础的系统需求分析. 广州: 中山大学软件研究所, 2002.
- [23] 刘超, 张莉. 可视化面向对象建模技术——标准建模语言 UML 教程. 北京: 北京航空航天大学出版社, 1999.
- [24] 洪永清, 黄德才, 吕丽民. 面向对象建模与设计. 北京: 人民邮电出版社, 1998.