****

****

**本 科 毕 业 设 计**

**院 系** 软件学院

**专 业** 软件工程

**题 目** 智造链系统财务部和物流部模块的设计与实现

**年 级** 2010级 **学 号** 101250108

**学生姓名** 莫其凡

**指导教师** 葛季栋 **职 称** 讲师

**论文提交日期**

**南京大学本科生毕业论文（设计）中文摘要**

毕业论文题目： 智造链系统财务部和物流部模块的设计与实现

软件学院 院系 软件工程 专业 2010 级本科生姓名： 莫其凡

指导教师（姓名、职称）： 葛季栋 讲师

摘要：

在初期的服装制造供应链中，各部门人员依靠记忆、口头交流、纸质文件进行工作信息的交流和记录，这样的生产过程没有条序，缺乏管理，在企业初期或许能够较为妥善的完成生产供应任务，但是随着企业和生产规模的扩大，必然会产生管理上的混乱，对未来的业务扩展造成瓶颈。

智造链系统按照不同部门不同角色划分模块，将各部门不同角色在各阶段的任务分类展示，使用BPMN2流程图将复杂的流程图形化使之清晰明了，一方面提高了生产供应链的工作效率和管理力度，提供给管理人员监控流程进度的可视化界面，另外一方面也为公司以后的业务信息化以及义务扩展奠定基础。

智造链系统采用B/S架构，前台使用HTML5+CSS3+Ajax结合Bootstrap+jQuery框架，后台使用JavaEE结合Spring+Hibernate+jBPM框架开发。jBPM是jBoss下的一款工作流引擎，用于管理流程的开始、跳转与结束，简化了开发人员的工作，使得我们能够更加专注于业务功能模块的实现。

制造链项目组目前完成了八个部门，包括人事部、市场部、设计部、采购部、生产部、物流部、财务部、质检部的基本功能模块的实现。

本人在项目组中负责完成财务部和物流部模块的设计开发与测试工作，同时负责整个项目业务流程相关的设计工作。

关键词：工作流、智造链、jBPM、物流部、财务部。

**南京大学本科生毕业论文（设计）英文摘要**

THESIS: Design and implementation of finance department module and logistics department module in FMC System

DEPARTMENT: Software Institute

SPECIALIZATION: Software Engineering

UNDERGRADUATE: 2010

MENTOR: Jidong Ge

ABSTRACT:

In early days in the fmc, people in different departments communicate with each other for work information through memory, verbal communication and paper materials, which is a process lack of good management. Maybe in the beginning of the development, the produce task may be completed well. However, with the company growing, there will be a mess in the management of the process and it will restrict the development of the business.

FMC System divide different people in their department into different module, provides respective user interfaces for each one in different phase, using BPMN2.0 diagram to make the whole process clear. On one hand, it improves the worker’s efficiency, gives the manager a visual interface to monitor the process, on the other hand, it helps the company make information transformation and will lay a good foundation for the future development.

FMC System uses B/S structure, the front end are based on HTML5, CSS3, Ajax along with Bootstrap and jQuery framework. The server end uses J2EE, Spring, Hibernate and jBPM framework which is a java business process management framework under jBoss. jBPM can help control the beginning and end of the process, making our work easier so that we can focus on the business logistics module.

FMC System includes eight departments, human resources department, market department, design department, purchase department, produce department, logistics department, financial department, quality inspection department. The basic module of the eight departments has been completed.

In the FMC System, I’m responsible for the financial department module and the logistics department module, also the whole process module.

KEYWORD: workflow, FMC, jBPM, finance ,logistics department.

目录

[图目录 III](#_Toc389595050)

[表目录 V](#_Toc389595051)

[第一章 引言 1](#_Toc389595052)

[1.1 项目背景 1](#_Toc389595053)

[1.2 工作流技术的发展历史和现状 1](#_Toc389595054)

[1.3 论文的主要工作和组织结构 2](#_Toc389595055)

[第二章 智造链系统技术概述 4](#_Toc389595056)

[2.1 Spring 4](#_Toc389595057)

[2.1.1 Spring简介 4](#_Toc389595058)

[2.1.2 Spring模块 4](#_Toc389595059)

[2.2 Hibernate 5](#_Toc389595060)

[2.2.1 Hibernate简介 5](#_Toc389595061)

[2.2.2 Hibernate接口 6](#_Toc389595062)

[2.3 jBPM 6](#_Toc389595063)

[2.3.1 jBPM简介 6](#_Toc389595064)

[2.3.2 BPMN简介 7](#_Toc389595065)

[2.4 github 7](#_Toc389595066)

[2.5 jQuery 8](#_Toc389595067)

[2.6 本章小结 8](#_Toc389595068)

[第三章 智造链系统需求分析与概要设计 9](#_Toc389595069)

[3.1 智造链系统整体概述 9](#_Toc389595070)

[3.1.1 系统用户角色 9](#_Toc389595071)

[3.1.2 系统功能性需求 9](#_Toc389595072)

[3.1.3 系统非功能性需求 10](#_Toc389595073)

[3.2 智造链系统财务部、物流部模块的需求分析 11](#_Toc389595074)

[3.2.1 流程图 11](#_Toc389595075)

[3.2.2 用例图 12](#_Toc389595076)

[3.2.3 用例描述 13](#_Toc389595077)

[3.2.4 实体关系 18](#_Toc389595078)

[3.3 智造链系统财务部、物流部模块的概要设计 18](#_Toc389595079)

[3.3.1 结构设计 19](#_Toc389595080)

[3.3.2 模块划分 20](#_Toc389595081)

[3.3.3 接口设计 20](#_Toc389595082)

[3.4本章小结 23](#_Toc389595083)

[第四章 智造链系统财务部、物流部模块的详细设计与实现 24](#_Toc389595084)

[4.1 财务部、物流部模块概述 24](#_Toc389595085)

[4.1.1 财务部模块概述 24](#_Toc389595086)

[4.1.2 物流部模块概述 24](#_Toc389595087)

[4.2 财务部、物流部模块的详细设计 24](#_Toc389595088)

[4.2.1 详细类图 24](#_Toc389595089)

[4.2.2 顺序图 26](#_Toc389595090)

[4.2.3 相关数据表 28](#_Toc389595091)

[4.3 财务部、物流部模块的实现 30](#_Toc389595092)

[4.3.1样衣金确认 30](#_Toc389595093)

[4.3.2首定金确认 32](#_Toc389595094)

[4.3.3尾款金确认 33](#_Toc389595095)

[4.3.4样衣收取 33](#_Toc389595096)

[4.3.5样衣发货 34](#_Toc389595097)

[4.3.6产品入库 35](#_Toc389595098)

[4.3.7产品发货 38](#_Toc389595099)

[4.4本章小结 40](#_Toc389595100)

[第五章 总结与展望 41](#_Toc389595101)

[5.1 总结 41](#_Toc389595102)

[5.2 展望 41](#_Toc389595103)

[参考文献 42](#_Toc389595104)

[致谢 43](#_Toc389595105)

# 图目录

[图2.1 Spring框架图 4](#_Toc389595143)

[图2.2 Hibernate结构图 5](#_Toc389595144)

[图2.3 jBPM结构图 7](#_Toc389595145)

[图2.4 jBPM流程示例图 7](#_Toc389595146)

[图3.1 智造链系统需求图 10](#_Toc389595147)

[图3.2 智造链系统流程图 11](#_Toc389595148)

[图3.3 财务部用例图 12](#_Toc389595149)

[图3.4 物流部用例图 13](#_Toc389595150)

[图3.5 智造链系统实体关系图 18](#_Toc389595151)

[图3.6 程序结构图 19](#_Toc389595152)

[图3.7 系统功能模块划分图 20](#_Toc389595153)

[图4.1 物流部模块详细类图 25](#_Toc389595154)

[图4.2 财务部模块详细类图 26](#_Toc389595155)

[图4.3 物流主管入库顺序图 27](#_Toc389595156)

[图4.4 物流主管发货顺序图 28](#_Toc389595157)

[图4.5 财务主管样衣金确认任务列表图 31](#_Toc389595158)

[图4.6 FinanceController的confirmSampleMoneyList方法表 31](#_Toc389595159)

[图4.7 ServiceUtil的getOrderList方法表 32](#_Toc389595160)

[图4.8 财务主管样衣金确认任务详情图 32](#_Toc389595161)

[图4.9 财务主管首定金确认任务详情图 33](#_Toc389595162)

[图4.10 物流主管样衣收取任务列表图 33](#_Toc389595163)

[图4.11 物流主管样衣收取任务详情图 34](#_Toc389595164)

[图4.12 物流主管样衣发货任务详情图 34](#_Toc389595165)

[图4.13 物流主管产品入库任务列表图 35](#_Toc389595166)

[图4.14 物流主管装箱任务详情图 35](#_Toc389595167)

[图4.15 物流主管装箱打印任务详情图 36](#_Toc389595168)

[图4.16 物流主管装包单图 36](#_Toc389595169)

[图4.17 条形码生产方法表 36](#_Toc389595170)

[图4.18 物流主管扫描设备任务图 37](#_Toc389595171)

[图4.19 物流主管扫描入库任务列表图 37](#_Toc389595172)

[图4.20 扫描监听方法表 37](#_Toc389595173)

[图4.21 物流主管扫描入库任务详情图 38](#_Toc389595174)

[图4.22 物流主管扫描设备任务图 38](#_Toc389595175)

[图4.23 物流主管发货扫描任务列表图 39](#_Toc389595176)

[图4.24 物流主管发货扫描任务详情图 39](#_Toc389595177)

[图4.25 物流主管产品发货任务列表图 40](#_Toc389595178)

[图4.26 物流主管产品发货任务详情图 40](#_Toc389595179)

# 表目录

[表3.1 样衣费确认用例表 13](#_Toc389595180)

[表3.2 首定金确认用例表 14](#_Toc389595181)

[表3.3 尾款金确认用例表 15](#_Toc389595182)

[表3.4 样衣收取用例表 15](#_Toc389595183)

[表3.5 样衣发货用例表 16](#_Toc389595184)

[表3.6 产品入库用例表 16](#_Toc389595185)

[表3.7 产品发货用例表 17](#_Toc389595186)

[表3.8 FinanceService接口方法表 21](#_Toc389595187)

[表3.9 IMoneyDao接口方法表 21](#_Toc389595188)

[表3.10 LogisticsService接口方法表 21](#_Toc389595189)

[表3.11 ILogisticsDao接口方法表 22](#_Toc389595190)

[表3.12 IPackageDao接口方法表 22](#_Toc389595191)

[表3.13 IPackageDetailDao接口方法表 23](#_Toc389595192)

[表4.1 汇款信息表 28](#_Toc389595193)

[表4.2 物流信息表 29](#_Toc389595194)

[表4.3 装箱信息表 30](#_Toc389595195)

[表4.4 箱子详情信息表 30](#_Toc389595196)

# 第一章 引言

## 1.1 项目背景

服装供应链是一个传统并且复杂的过程，其中涉及的人员角色包括市场部、设计部、采购部、生产部、物流部、人事部、财务部、质检部，甚至每个部门也会有不同的角色分工，在整个过程中，涉及各种角色工作信息之间的交流和沟通，环环相扣，紧密联系。

然而现在该公司的管理力度不高，各个角色的工作甚至只通过脑力记忆和口头交流和少量的纸质文件，对于现阶段小规模的生产勉强能够应付，但是随着企业业务规模的壮大，必然不能满足企业未来的发展需求，会出现各种供应链环节的管理混乱问题。在如今这个信息化互联网普及转型的阶段，使用一套软件来规范管理供应链流程显得十分必要。不仅能够使得人员分工更加清晰，企业人员更加好的监控业务流程进度，提高工作效率，同时，信息化的转型使得未来企业的发展奠定一个良好的基础。

工作流的概念起源于生产组织和办公自动化领域，主要针对在日常的工作之中拥有固定程序的活动而提出的一个概念，为的是通过将工作按照一定的任务和角色进行划分，同时按照某种规则来执行这些任务并对其进行监控，以此使得工作效率提高、更好的监控管理业务流程[1]。根据工作流管理联盟（WfMC）的定义：工作流是业务过程的整体或者部分在计算机环境下的自动化[2]，使得各个参与者按照某种规则传递信息，促使业务过程自动进行从而完成业务目标[3]。

本文介绍的智造链FMC系统，以jBPM工作流框架为基础，将整个服装供应链流程良好的组织起来，将各个角色的任务明确罗列，提高了各个部门的效率，不仅给管理者提供一个流程监控的可视化平台，同时为企业未来提供信息数据化分析创造了良好的基础。

## 1.2 工作流技术的发展历史和现状

工作流技术起源于20世纪70年代中期的办公自动化领域。在当时与工作流相关的研究工作有SCOOP原型系统以及OfficeTalk研究系统等，它们都使用了某一种Petri的变体对流程进行建模，是最早的办公自动化系统。然而当时的工作流技术迫使人们必须按照某些固定的流程进行，非常死板，缺乏灵活性，人们往往希望能够在某些恰当的时机打破某些标准的束缚，因此当时的办公效率有时会很低效，同时由于个人计算机尚未普及，互联网技术等限制，工作流技术的发展并没有取得预期的效果[4][5]。

1983年到1985年的时间内，早起的商用工作流系统被开发出来，它们主要用于电子邮件业务以及图像处理领域。图像处理中的流转以及跟踪、电子邮件的流程流转都采用了工作流的思想[4]。

随着相关技术的成熟，进入20世纪90年代，工作流系统的发展进入了新的阶段，制造业、软件工程、银行业、电讯业等领域都开始使用工作流技术。工作流管理联盟（WfMC）于1993年成立，该组织创建完善了与工作流相关的一系列标准，是致力于工作流标准化的专业组织。此外该组织还推出了工作流XML和XML过程定义语言[4]。

21世纪以后，伴随Web Service技术的兴起，许多标准化组织各自制定了与工作流相关的Web Serive标准，例如WSFL（IBM制定）和XLANG。2002年，微软、IBM和BEA开发了BPEL（Business Process Execution Language）语言。BPEL基于XML，用于描写业务流程，业务流程的单个步骤由Web Service实现[4]。

如今一些比较常用的工作流框架包括jBPM、OSWorkflow、Activiti、Shark等。本文使用的工作流引擎是jBPM5.3。jBPM5完全抛弃了jBPM4的基础代码，基于Drools Flow重新开始，而原来jBPM的创建者Tom Baeyens离开JBoss, 加入Alfresco之后，推出了基于jBPM4的开源工作流系统Activiti[6]。

## 1.3 论文的主要工作和组织结构

本文主要介绍智造链FMC系统的工作流框架和其中相应的物流部门以及财务部门模块的需求、设计、开发等内容。整个智造链系统包括八个部门模块：市场部、人事部、设计部、采购部、生产部、质检部、财务部、物流部。本人主要完成了财务部和物流部两个模块的设计开发工作，以及整个系统工作流框架的设计使用。

第一章：引言部分，主要介绍了项目背景，工作流技术的发展历史和现状，并且简述了本片论文的主要工作和组织结构。

第二章：智造链系统技术概述，主要介绍了该系统设计开发过程中用到的主要技术，包括Spring框架、Hibernate数据持久化框架、jBPM工作流框架、git版本开发管理工具、jQuery等技术。

第三章：智造链系统需求分析与概要设计，主要介绍从需求和概要设计方面介绍了智造链系统的整体需求和设计架构，从用户角度和高层设计角度描述了系统的结构。

第四章：智造链系统财务部、物流部模块的详细设计与实现，主要介绍了八个部门中财务部和物流部门模块的详细设计与实现，描述了系统使用的细节过程。

第五章：总结与展望，对制造链系统整个项目进行了总结，描述整个项目的完成情况和进度，以及自己在项目组中的工作，同时，对未来系统进一步的扩展和完善做一定的规划。

# 第二章 智造链系统技术概述

## 2.1 Spring

### 2.1.1 Spring简介

Spring是由Rod Johnson创建的一个开源的轻量级框架，它通过loC控制反转降低耦合，减少类之间的相互依赖，同时使用面向切面AOP的思想，将业务逻辑从系统服务中分离出来，提高模块的内聚化，将服务模块化。使用配置文件和代码注解的方式实现依赖注入[7]。如图2.1所示为Spring的框架图。



图2.1 Spring框架图

### 2.1.2 Spring模块

组成 Spring 框架的模块可以单独存在，或者与其他的模块协作实现。每个模块的功能分别如下所示：

* 核心容器： Spring框架的基本功能由核心容器提供。BeanFactory是核心容器的主要组件，这是一种工厂模式的实现。依靠控制反转（loC），BeanFactory将程序的依赖性和配置与程序代码分开。
* Spring 上下文：Spring 上下文是一个配置文件，为Spring框架提供配置信息。Spring上下文包括企业服务，比如电子邮件、EJB、国际化、JNDI、校验和调度功能。
* Spring AOP：通过配置管理特性，Spring AOP 模块直接将面向方面的编程功能集成到了 Spring 框架中。所以，可以很容易地使 Spring 框架管理的任何对象支持 AOP。Spring AOP 模块为基于 Spring 的应用程序中的对象提供了事务管理服务。通过使用 Spring AOP，不用依赖 EJB 组件，就可以将声明性事务管理集成到应用程序中。
* Spring DAO：JDBC DAO 抽象层提供了有意义的异常层次结构，可用该结构来管理异常处理和不同数据库供应商抛出的错误消息。异常层次结构简化了错误处理，并且极大地降低了需要编写的异常代码数量（例如打开和关闭连接）。Spring DAO 的面向 JDBC 的异常遵从通用的 DAO 异常层次结构。
* Spring ORM：Spring 框架插入了若干个 ORM 框架，从而提供了 ORM 的对象关系工具，其中包括 JDO、Hibernate 和 iBatis SQL Map。所有这些都遵从 Spring 的通用事务和 DAO 异常层次结构。
* Spring Web 模块：Web 上下文模块建立在应用程序上下文模块之上，为基于 Web 的应用程序提供了上下文。所以，Spring 框架支持与 Jakarta Struts 的集成。Web 模块还简化了处理多部分请求以及将请求参数绑定到域对象的工作。
* Spring MVC 框架：MVC 框架是一个全功能的构建 Web 应用程序的 MVC 实现。通过策略接口，MVC 框架变成为高度可配置的，MVC 容纳了大量视图技术，其中包括 JSP、Velocity、Tiles、iText 和 POI[8]。

## 2.2 Hibernate

### 2.2.1 Hibernate简介

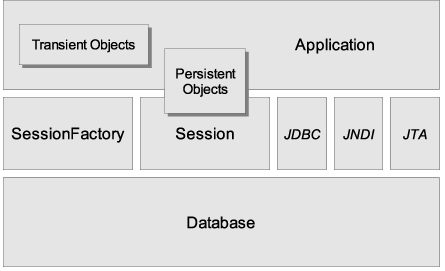


图2.2 Hibernate结构图

Hibernate是一款开源的对象关系映射ORM框架，使用面向对象的思想操作数据库，用于实现数据持久化的良好解决方案。Hibernate没有侵入性，利于移植，简化了开发人员手动处理SQL的工作[9]。如图2.2所示为Hibernate的结构图。

### 2.2.2 Hibernate接口

Hibernate一共有6个核心接口，分别是Session、SessionFactory、Criteria、Query、Transaction和Configuration。6个接口的主要功能介绍如下。

* Session：主要负责执行被持久化对象的CRUD操作。Session对象是非线程安全的。
* SessionFactory：负责初始化Hibernate，并创建Session对象。这是工厂模式的一种使用。
* Criteria：负责创建并执行面向对象的标准化查询。Query接口是轻量级的，不能在Session之外使用。
* Query：对数据库及持久对象进行查询，有两种表达方式：HQL语言和本地数据库的SQL语句。
* Transaction：负责事务处理的相关操作。
* Configuration：主要负责对Hibernate 进行配置，以及对它进行启动。在Hibernate 启动之后，会负责创建一个SessionFactory对象。

## 2.3 jBPM

### 2.3.1 jBPM简介

jBPM，即java business process management（java业务流程管理），目前是JBoss下的一款开源的轻量级工作流框架，jBPM核心引擎可以内嵌入你的应用，同时也支持作为一项服务部署。如图2.3所示为jBPM的结构图。

jBPM5.3使用BPMN2.0作为其流程图知识库，同时支持Spring+Hibernate框架结合。BPMN2.0定义的流程图主要包括事件（event）、活动（activity）、分支（gateway）等一系列元素[10] [11]。

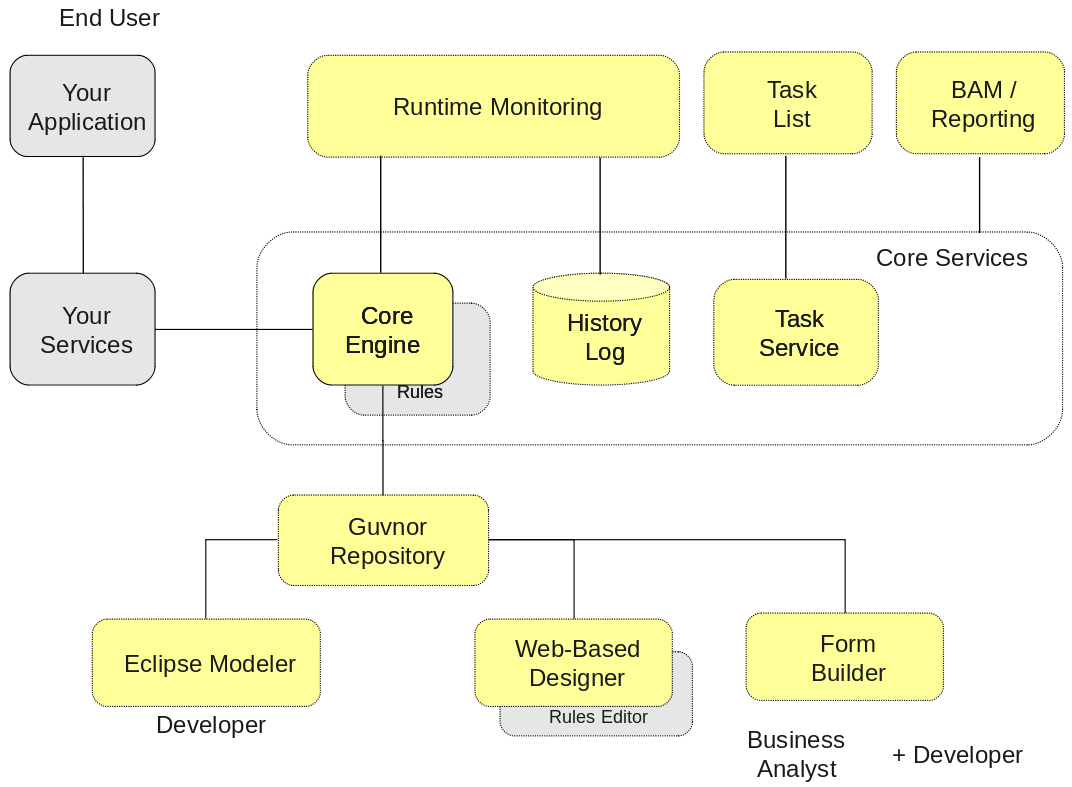


图2.3 jBPM结构图

### 2.3.2 BPMN简介

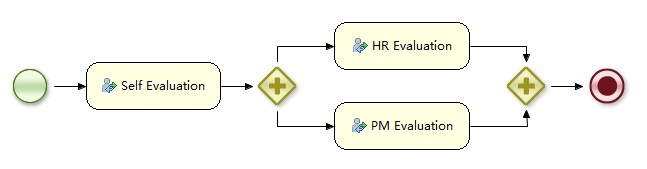


图2.4 jBPM流程示例图

BPMN即业务流程建模与标注（Business Process Model and Notation）。主要用于可视化的业务流程建模，同时适用于开发人员和业务人员的工作。

如图2.4所示，绿色圆圈表示流程开始事件，红色圆圈表示流程结束事件，图中的圆角矩形表示人工任务，是由角色手工完成的任务，圆角矩形还有其他类别，比如说脚本任务以及服务任务等，菱形表示分支（gateway），根据逻辑条件的判定决定流程的走向，可以是并发、选择、汇聚等多种类型。

## 2.4 github

git是一个分布式的版本控制系统，最初是由Linus编写实现，主要用于Linux内核代码的管理。区别于CVS和SVN那样的集中式版本控制系统，git使得用户能够在本地电脑拥有自己的代码库，使团队成员协同开发更加高效[12]。

## 2.5 jQuery

jQuery是在prototype之后又一个非常优秀的JavaScript框架。它不仅优化了遍历HTML文档、操作DOM、处理事件的操作，同时还实现了动画效果，优化Ajax操作。jQuery还有许多优点，比如它是轻量级的，拥有强大的选择器，出色的浏览器兼容性，链式操作方式，丰富的插件和文档支持等[13]

## 2.6 本章小结

本章主要介绍了智造链FMC项目系统中使用的主要技术，描述了项目开发过程后台使用的Spring+Hibernate框架、前台使用的jQuery框架，以及分布式版本控制工具git，同时简要介绍了系统核心的工作流框架jBPM。

# 第三章 智造链系统需求分析与概要设计

## 3.1 智造链系统整体概述

智造链系统是一款用于管理某公司服装生产供应链的工作流管理软件，一方面合理组织各个部门的工作进度流程，是复杂的流程变得清晰简明，提供可视化的流程监控界面，提高工作效率，另外一方面为企业的信息化管理奠定基础，为将来业务扩展，信息分析等做好准备。

### 3.1.1 系统用户角色

智造链系统目前主要包括八个部门，十种角色，分别为市场专员、市场主管、设计主管、采购主管、生产主管、财务主管、物流主管、人事主管以及管理员，除了市场专员为多个以外，各部门主管只有一人。除此之外未来可能会为高层管理人员提供功能模块。

### 3.1.2 系统功能性需求

根据部门和职责，将智造链系统的需求分为以下10个部分。

客户管理：包括新建客户、查看客户等主要功能。

订单管理：包括查看订单信息、查看订单进度等主要功能。

市场部：包括客户下单、修改询单、合并报价、审核报价、报价商定、修改报价、商定合同、修改合同、签订合同、订单回访、提醒缴费等主要功能。

设计部：包括设计验证、样衣版型、生产验证、生产版型等主要功能。

采购部：包括采购验证、成本核算、样衣采购、采购确认、生产采购等主要功能。

生产部：包括生产验证、成本核算、样衣生产、批量生产等主要功能。

财务部：包括样衣费确认、首定金确认、尾款金确认等主要功能。

物流部：包括样衣收取、样衣发货、产品入库、产品发货等主要功能。

质检部：包括质量检查等主要功能。

人事部：添加员工、查看员工等主要功能。

智造链系统需求划分如图3.1所示：



图3.1 智造链系统需求图

### 3.1.3 系统非功能性需求

易使用性

系统具有良好的人机交互特性，提供各种人性化的提示信息。

可靠性

系统的故障频率控制在3个月/次。

系统发生故障后能在24小时内修复。

可扩展性

在未来能为高层管理者提供管理决策层次上的分析等功能。

安全性

系统要能够保证客户信息以及业务信息的安全性，不被他人窃取。

可用性：

系统能在用户操作后1秒内及时反馈。

## 3.2 智造链系统财务部、物流部模块的需求分析

### 3.2.1 流程图

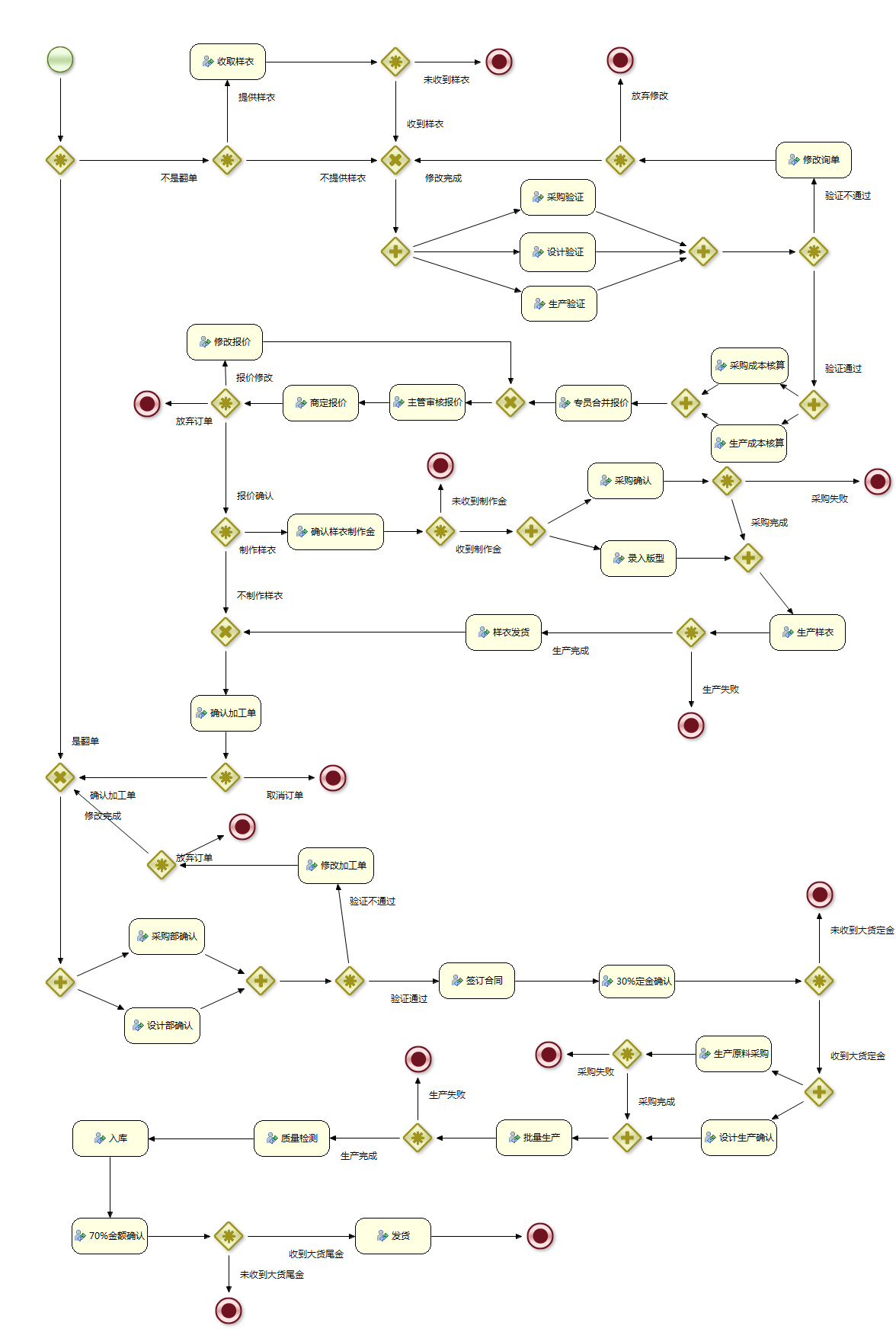


图3.2 智造链系统流程图

如图3.2所示，智造链系统的生命周期从下单开始，最终到发货结束整个流程，其中还有各种分支，图中采用圆角矩形表示人工任务，菱形表示门（gateway），包括条件分支和汇聚。

### 3.2.2 用例图



图3.3 财务部用例图

如图3.3所示，财务部主管有三个主要的用例，分别为样衣费用确认、首定金确认以及尾款金确认。

其中样衣费用=样衣件数\*制版费用；根据衣服类型划分，制版费用分别为200元（春夏款）和400元（秋冬款）。

首定金为总金额的30%，尾款金为70%，其中总金额=（样衣件数+大货件数）\*单件客户报价-样衣费用-优惠金额。

如图3.4所示，物流部主管主要有四个用例，分别为样衣收取、样衣发货、产品入库，产品发货。其中产品入库又包括产品装箱和扫描入库两个步骤，产品发货包括扫描发货和发货登记两个步骤。



图3.4 物流部用例图

### 3.2.3 用例描述

财务部

财务部模块主要包括3个用例：分别是样衣费确认、首定金确认和尾款金确认。具体用例描述分别如下表3.1、表3.2、表3.3所示。

表3.1 样衣费确认用例表

|  |  |
| --- | --- |
| 用例编号 | 01 |
| 用例名称 | 样衣费确认 |
| 用例描述 | 在样衣生产完成之后确认样衣制作费用，填写相关收款信息 |
| 触发条件 | 样衣生产完成 |
| 前置条件 | 登陆身份为财务主管 |
| 后置条件 | 寄送样衣 |
| 用例过程 | 1. 财务主管登陆网站 2. 点击财务部/样衣金确认菜单 3. 查看任务列表并选择某一任务点击详情 4. 查看订单信息，包括基本信息、面辅信息、样衣信息、加工信息、版型信息等内容 5. 在线下确认收款之后在系统内填写收款信息，包括汇款人姓名、汇款账号、汇款银行、汇款金额、收款金额和备注信息，点击按钮提交。 6. 跳转会任务列表 |
| 扩展描述 | 5a、输入信息不完整或者数据类型错误时给出提示信息 |
| 备注信息 | 无 |

表3.2 首定金确认用例表

|  |  |
| --- | --- |
| 用例编号 | 02 |
| 用例名称 | 首定金确认 |
| 用例描述 | 在签订大货合同之后确认大货定金，填写相关收款信息 |
| 触发条件 | 签订大货合同完成 |
| 前置条件 | 登陆身份为财务主管 |
| 后置条件 | 采购设计部确认 |
| 用例过程 | 1. 财务主管登陆网站 2. 点击财务部/首定金确认菜单 3. 查看任务列表并选择某一任务点击详情   4、查看订单信息，包括基本信息、面辅信息、样衣信息、加工信息、版型信息等内容  5、在线下确认收款之后在系统内填写收款信息，包括汇款人姓名、汇款账号、汇款银行、汇款金额、收款金额和备注信息，点击按钮提交。  6、跳转会任务列表 |
| 扩展描述 | 5a、输入信息不完整或者数据类型错误时给出提示信息 |
| 备注信息 | 无 |

表3.3 尾款金确认用例表

|  |  |
| --- | --- |
| 用例编号 | 03 |
| 用例名称 | 尾款金确认 |
| 用例描述 | 在大货入库合同之后确认大货尾款，填写相关收款信息 |
| 触发条件 | 大货入库 |
| 前置条件 | 登陆身份为财务主管 |
| 后置条件 | 物流发货 |
| 用例过程 | 1. 财务主管登陆网站 2. 点击财务部/尾款金确认菜单   3、查看任务列表并选择某一任务点击详情  4、查看订单信息，包括基本信息、面辅信息、样衣信息、加工信息、版型信息等内容  5、在线下确认收款之后在系统内填写收款信息，包括汇款人姓名、汇款账号、汇款银行、汇款金额、收款金额和备注信息，点击按钮提交。  6、跳转会任务列表 |
| 扩展描述 | 5a、输入信息不完整或者数据类型错误时给出提示信息 |
| 备注信息 | 无 |

物流部

物流部模块主要包括4个用例：分别是样衣收取、样衣发货、产品入库和产品发货。据图用例描述分别如下表3.4、表3.5、表3.6、表3.7所示。

表3.4 样衣收取用例表

|  |  |
| --- | --- |
| 用例编号 | 04 |
| 用例名称 | 样衣收取 |
| 用例描述 | 收取客户提供的样衣，在系统内确认收取信息 |
| 触发条件 | 客户提供样衣 |
| 前置条件 | 登陆身份为物流主管 |
| 后置条件 | 采购设计生产部验证 |
| 用例过程 | 1. 物流主管登陆网站 2. 点击物流部/样衣收取菜单 3. 查看任务列表并选择某一任务点击详情 4. 查看订单信息，包括基本信息、面辅信息、样衣信息、加工信息、版型信息等内容 5. 在线下收取样衣之后对比样衣信息，确认后点击已收取样衣确认   6、跳转会任务列表 |
| 扩展描述 | 4a、未收取到样衣则点击未收到样衣按钮 |
| 备注信息 | 无 |

表3.5 样衣发货用例表

|  |  |
| --- | --- |
| 用例编号 | 05 |
| 用例名称 | 样衣发货 |
| 用例描述 | 样衣生产完成后物流部寄送样衣并且填写发货信息 |
| 触发条件 | 生产部完成样衣生产 |
| 前置条件 | 登陆身份为物流主管 |
| 后置条件 | 市场部确认生产加工单 |
| 用例过程 | 1. 物流主管登陆网站 2. 点击物流部/样衣发货菜单 3. 查看任务列表并选择某一任务点击详情   4、 查看订单信息，包括基本信息、面辅信息、样衣信息、加工信息、版型信息等内容  5、在线下样衣发货之后填写物流信息，包括邮寄时间、快递名称、快递单号和备注信息，点击完成发货按钮  6、跳转会任务列表 |
| 扩展描述 | 5a、物流信息填写不完整或者类型错误时给出提示信息 |
| 备注信息 | 无 |

表3.6 产品入库用例表

|  |  |
| --- | --- |
| 用例编号 | 06 |
| 用例名称 | 产品入库 |
| 用例描述 | 大货质检之后将其装箱，填写装箱信息，贴上条形码，扫描入库，填写入库信息 |
| 触发条件 | 大货完成质检 |
| 前置条件 | 登陆身份为物流主管 |
| 后置条件 | 大货尾款金确认 |
| 用例过程 | 1、物流主管登陆网站  2、点击物流部/产品入库菜单  3、查看任务列表并选择某一任务点击详情  4、查看订单信息，包括基本信息、面辅信息、样衣信息、加工信息、版型信息等内容  5、线下完成装箱后填写具体装箱信息，包括每箱的衣服的颜色、大小、数量，填写完成后点击完成装箱  6、系统跳转到条形码打印页面，依次打印每箱的条形码贴在箱子上  7、将产品入库，使用手持设备登陆网站  8、点击产品入库，选择对应订单号，依次扫描箱子上的条形码，并且填写相应箱子存放的仓库、货架、具体位置。  9、扫描完毕后点击完成入库 |
| 扩展描述 | 5a、填写数据为空或者错误类型则显示相应的提示信息  8a、没有将订单相关的全部箱子入库信息扫描则显示相应提示信息  8b、扫描了不属于该订单的箱子则提示相应信息 |
| 备注信息 | 无 |

表3.7 产品发货用例表

|  |  |
| --- | --- |
| 用例编号 | 07 |
| 用例名称 | 产品发货 |
| 用例描述 | 大货尾款确认之后物流发货，填写相关发货信息 |
| 触发条件 | 大货尾款确认 |
| 前置条件 | 登陆身份为物流主管 |
| 后置条件 | 订单回访 |
| 用例过程 | 1. 物流主管登陆网站 2. 点击物流部/产品发货菜单 3. 查看任务列表并选择某一任务点击详情、 4. 查看订单信息，包括基本信息、面辅信息、样衣信息、加工信息、版型信息等内容 5. 使用手持设别登陆网站 6. 点击产品大货并选择相应订单号 7. 依次扫描待发货箱子条形码确认箱包并点击确认 8. 线下完成发货之后登陆网站 9. 填写发货信息包括发货时间、快递名称、快递单号、快递费用、备注信息等、点击提交按钮确认。 |
| 扩展描述 | 6a、未扫描订单的全部则显示相应提示信息  6b、扫描的箱子不属于该订单则显示相应提示信息 |
| 备注信息 | 无 |

### 3.2.4 实体关系

每个询单有多条收款信息，所以询单和收款信息关系为1:n。

每个询单有一条物流信息，所以询单和物流信息的关系为1:1。

每个询单有多条装箱信息，所以询单和装箱信息关系为1:n。

每个装箱信息有多条装箱详情记录，装箱信息与装箱详情记录关系为1:n。

如图3.5所示为与物流部和财务部相关的主要ERD，其他实体的属性信息由于数量比较多并且与财务部、物流部无关所以在此省略。



图3.5 智造链系统实体关系图

## 3.3 智造链系统财务部、物流部模块的概要设计

本章主要介绍智造链系统的结构设计、组件模块划分以及财务部、物流部相关接口的设计。

### 3.3.1 结构设计

智造链系统的开发设计将分为五个层次，分别是View（视图）层、Controller（控制器）层、Service（服务）层、DAO（数据访问）层和DB（数据库）层。使用Spring注解和SpringMVC将前端页面、业务逻辑、数据持久化分离开来，一方面降低了耦合，提高了模块的内聚性，使得开发人员能够更好的并行开发。



图3.6 程序结构图

智造链系统的主要结构设计如图3.6所示。系统用户通过与View交互，View发送请求给Controller，Controller将请求转发给对应的Service服务，Service服务调用相关Dao完成业务逻辑，Dao调用DB接口完成相应的数据持久化与更新操作，ServiceImpl实现了Service接口，DaoImpl实现了Dao接口，同时Service使用JBPMUtil完成业务流程的控制，使用ServiceUtil获得流程相关的业务信息。

### 3.3.2 模块划分

智造链系统由六个模块组成，如图3.7所示，分别为数据持久化组件、订单管理组件、流程管理组件、用户管理组件、文件管理组件以及权限管理组件组成。

物流部和财务部主要涉及其中的数据持久化组件、订单管理组件、流程管理组件以及权限管理组件。



图3.7 系统功能模块划分图

### 3.3.3 接口设计

智造链系统财务部和物流部的主要接口涉及Service和Dao层。财务部接口主要有FinanceService与IMoneyDao。物流部主要接口有LogisticsService、ILogisticsDao、IPackageDao和IPackageDetailDao。

FinanceService接口由FinanceServiceImpl类实现，其接口定义如表3.8所示。该接口主要提供了为财务主管获得三类金额确认任务列表与详情的方法接口以及提交金额确认信息的方法接口。

表3.8 FinanceService接口方法表

|  |  |
| --- | --- |
| 方法名 | 描述 |
| getConfirmSampleMoneyList | 获得样衣金确认任务列表 |
| getConfirmSampleMoneyDetail | 获得样衣金确认任务详情 |
| confirmSampleMoneySubmit | 提交样衣金确认信息 |
| getConfirmDepositList | 获得首定金确认任务列表 |
| getConfirmDepositDetail | 获得首定金确认任务详情 |
| confirmDepositSubmit | 提交首定金确认信息 |
| getConfirmFinalPaymentList | 获得尾款金确认任务列表 |
| getConfirmFinalPaymentDetail | 获得尾款金确认任务详情 |
| confirmFinalPaymentSubmit | 提交尾款金确认信息 |

IMoneyDao接口由MoneyDao类实现，其接口定义如表3.9所示，该接口主要提供了对收款金额信息的增删查改的方法。

表3.9 IMoneyDao接口方法表

|  |  |
| --- | --- |
| 方法名 | 描述 |
| save | 添加收款信息 |
| findById | 根据id查找收款信息 |
| findByOrderId | 根据订单id查找收款信息 |
| findByProperty | 根据属性查找收款信息 |
| attachDirty | 更新收款信息 |
| delete | 删除收款信息 |

LogisticsService接口由LogisticsServiceImpl类实现，其接口定义由表3.10所示。该接口主要提供了为物流主管获得四类任务（收取样衣、样衣发货、大货入库、大货发货）列表和详情的方法接口，以及提交样衣收货信息、样衣发货信息，大货装箱入库信息和大货发货信息的方法接口。

表3.10 LogisticsService接口方法表

|  |  |
| --- | --- |
| 方法名 | 描述 |
| getReceiveSampleList | 获得样衣收取任务列表 |
| getReceiveSampleDetail | 获得样衣收取任务详情 |
| receiveSampleSubmit | 提交样衣收取信息 |
| getSendSampleList | 获得样衣发货任务列表 |
| getSendSampleDetail | 获得样衣发货任务详情 |
| sendSampleSubmit | 提交样衣发货信息 |
| getPackageList | 获得装箱任务列表 |
| getWarehouseList | 获得入库任务列表 |
| getPackageDetail | 获得装箱任务详情 |
| getPrintWarehouseDetail | 获得入库打印单详情 |
| addPackage | 添加箱子信息 |
| removePackage | 删除箱子信息 |
| packageSubmit | 提交装箱信息 |
| getMobileWarehouseList | 获得装箱入库扫描任务列表 |
| getMobileWarehouseDetail | 获得装箱入库扫描任务详情 |
| updatePackage | 更新箱子信息 |
| mobileWarehouseSubmit | 提交入库信息 |
| getScanClothesList | 获得待扫描发货任务列表 |
| getSendClothesList | 获得发货信息填写任务列表 |
| getSendClothesDetail | 获得发货信息填写任务详情 |
| getMobileSendClothesList | 获得待发货扫描任务列表（手持设备） |
| getMobileSendClothesDetail | 获得待发货扫描任务详情（手持设备） |
| mobileSendClothesSubmit | 提交扫描发货结果（手持设备） |
| sendClothesSubmit | 提交发货信息 |

ILogisticsDao由类LogisticsDao实现，其接口定义如表3.11所示。该接口主要提供了对物流信息的添加、修改和查找的方法。

表3.11 ILogisticsDao接口方法表

|  |  |
| --- | --- |
| 方法名 | 描述 |
| save | 添加物流信息 |
| findById | 根据id查找物流信息 |
| attachDirty | 更新物流信息 |

IPackageDao由类PackageDao实现，其接口定义如表3.12所示。该接口主要提供了对装箱信息的增删查改的方法。

表3.12 IPackageDao接口方法表

|  |  |
| --- | --- |
| 方法名 | 描述 |
| save | 添加装箱信息 |
| findById | 根据id查找装箱信息 |
| findByOrderId | 根据订单id查找装箱信息 |
| attachDirty | 更新装箱信息 |
| delete | 删除装箱信息 |

IPackageDetailDao由类PackageDetailDao实现，其接口定义如表3.13所示。该接口主要提供了对装箱详情信息的增删查的方法。

表3.13 IPackageDetailDao接口方法表

|  |  |
| --- | --- |
| 方法名 | 描述 |
| save | 添加装箱详情信息 |
| findByPackageList | 根据装箱列表查找装箱详情信息 |
| findByPackageId | 根据装箱id查找装箱详情信息 |
| delete | 删除装箱详情信息 |

## 3.4本章小结

本章内容首先介绍了智造链系统的整体概述，描述了系统整体上的功能，下面依次介绍了系统的需求分析，从用例图、用例描述、实体关系图等多个方面进行用例分析，紧接着介绍系统的概要设计，包括系统整体结构设计、模块划分以及物流部、财务部相应模块的接口设计。

# 第四章 智造链系统财务部、物流部模块的详细设计与实现

## 4.1 财务部、物流部模块概述

### 4.1.1 财务部模块概述

财务部模块在智造链系统中功能主要是提供给财务主管一个清晰的界面，为收取各种款项提供明细资料，同时允许财务主管填写相关的收取信息。

在一次询单中，财务主管要确认样衣金、大货定金、大货尾款的确认，系统必须提供询单的收款统计信息给财务主管，方面财务主管确认各类金额。

### 4.1.2 物流部模块概述

物流部模块在智造链系统中属于比较重要的一个环节，物流主管主要负责样衣的收取和发送、大货装箱入库、大货扫描发货等操作。

物流部的功能在整个系统中比较复杂，其中产品入库和产品发货部分涉及的流程比较多。产品入库又分为产品装箱和产品扫描，产品装箱的时候会生成条形码，用于入库阶段的扫描。产品发货包括扫描发货和发货登记，发货前也会扫描条形码确认，保证发货流程的可靠性以及安全性。

## 4.2 财务部、物流部模块的详细设计

### 4.2.1 详细类图

在智造链系统中，Controller负责响应前端请求，将其转发给Service对应的接口处理， Service利用IDAO和公用的流程模块JBPMUtil和ServiceUtil进行流程的操作以及数据库的持久化就更新操作。DAO实现IDAO接口，ServiceImpl实现Service接口。



图4.1 物流部模块详细类图

如图4.1所示，物流部模块涉及的类有LogisticsController、LogisticsService、LogisticsServiceImpl、ILogisticsDAO、LogisticsDAO、IPackageDao、PackageDao、IPackageDetailDao、PackageDetailDao。其中LogisticsController将请求转发给LogisticsService，LogisticsService处理业务逻辑同时调用LogisticsDAO、IPackageDao、IPackageDetailDao完成数据的持久化。在这里省略了公用模块JBPMUtil和ServiceUtil的类图。

如图4.2所示，财务部模块涉及的类有FinanceController、FinanceService、FinanceServiceImpl、IFinanceDAO、FinanceDAO。财务部基本情况与物流部相同，在此不做赘述。



图4.2 财务部模块详细类图

### 4.2.2 顺序图

在财务部和物流部的相关任务中，比较复杂的流程有两个，分别是物流主管的产品入库和产品发货两个阶段。

产品入库：如图4.3所示，物流主管先添加多个装箱信息，然后提交，接着一次打印每个箱子的大货装箱单，将箱子一次入库，挨个扫描箱子上的条形码，输入入库信息，完成入库。

产品发货：如图4.4所示，物流主管依次扫描所有箱子的条形码，扫描完毕后线下完成发货，登陆网站更新物流发货信息。



图4.3 物流主管入库顺序图



图4.4 物流主管发货顺序图

### 4.2.3 相关数据表

财务部主要涉及到的数据表包括汇款信息表，汇款信息表记录顾客的汇款详情，其中包括汇款类型、汇款人信息、收款人信息、汇款时间、金额等内容，同时由于一个订单包含多条汇款信息记录，所以表中包含外键orderId，汇款信息表具体字段内容如表4.1所示。

表4.1 汇款信息表

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 描述 |
| id | Integer | 主键，唯一编号 |
| orderId | Integer | 订单编号，外键 |
| moneyType | String | 汇款类型 |
| moneyState | String | 汇款状态 |
| moneyName | String | 汇款人姓名 |
| moneyBank | String | 汇款银行 |
| moneyNumber | String | 汇款卡号 |
| moneyRemark | String | 备注信息 |
| receiveTime | Timestamp | 收款时间 |
| receiveAccount | String | 收款账号 |
| moneyAmount | Double | 汇款金额 |

物流部主要涉及到的数据表包括物流信息表，装箱信息表以及箱子详情表。

每个订单有一条物流信息，记录包括样衣收取、样衣发货、大货发货在内的所有物流信息。物流信息表具体内容字段如表4.2所示。

表4.2 物流信息表

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 描述 |
| orderId | Integer | 订单编号 |
| inPostSampleClothesTime | String | 样衣（客户提供）寄送时间 |
| inPostSampleClothesType | String | 样衣（客户提供）寄送快递名称 |
| inPostSampleClothesNumber | String | 样衣（客户提供）寄送快递单号 |
| sampleClothesType | String | 样衣（公司提供）快递名称 |
| sampleClothesAddress | String | 样衣（公司提供）收货地址 |
| sampleClothesName | String | 样衣（公司提供）收件人名称 |
| sampleClothesPhone | String | 样衣（公司提供）收件人电话 |
| sampleClothesTime | Timestamp | 样衣（公司提供）寄送时间 |
| sampleClothesRemark | String | 样衣（公司提供）寄送备注信息 |
| sampleClothesNumber | String | 样衣（公司提供）快递单号 |
| productClothesType | String | 大货寄送快递名称 |
| productClothesAddress | String | 大货收货地址 |
| productClothesPrice | String | 大货快递价格 |
| productClothesNumber | String | 大货快递单号 |
| productClothesName | String | 大货收件人名称 |
| productClothesPhone | String | 大货收件人电话 |
| productClothesTime | Timestamp | 大货发货时间 |
| productClothesRemark | String | 大货发货备注信息 |

每个订单因为有多个箱子所以有多条装箱信息，其中记录箱子的编号，具体仓库货架位置等信息，装箱信息表具体内容字段如表4.3所示。

表4.3 装箱信息表

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 描述 |
| packageId | Integer | 箱子编号，主键 |
| orderId; | Integer | 订单编号，外键 |
| packageTime | Timestamp | 装箱时间 |
| warehouseId | String | 仓库编号 |
| shelfId | String | 货架编号 |
| location | String | 位置信息 |

每条装箱信息记录有多条装箱详情信息，记录了箱子中衣物的大小、颜色和数量。装箱详情信息表具体内容如表4.4所示。

表4.4 箱子详情信息表

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 描述 |
| id | Integer | 主键，唯一编号 |
| packageId; | Integer | 箱子编号 |
| clothesStyleName | String | 衣服型号 |
| clothesStyleColor | String | 衣服颜色 |
| clothesAmount | Integer | 衣服数量 |

## 4.3 财务部、物流部模块的实现

财务部模块主要包括样衣金确认、首定金确认、尾款金确认等操作。物流部模块主要包括样衣收取、样衣发货、产品入库、产品发货等操作。本节描述具体的实现细节，以及相关的界面截图。

### 4.3.1样衣金确认

如图4.5所示，左侧为财务主管的导航菜单，有三种任务，当选择样衣费确认后，右侧显示任务列表，有简要订单信息，点击详情链接后显示订单的详细内容标签页。



图4.5 财务主管样衣金确认任务列表图

如图4.7后台ServiceUtil中的getOrderList方法利用jbpmAPIUtil类获得对应的任务列表，然后利用相应的DAO填充订单信息，将内容返回给controller。如图4.6，controller将数据输入JSP页面返回给前台用户。

|  |
| --- |
| // ===================样衣金确认=================  @RequestMapping(value = "/finance/confirmSampleMoneyList.do")  @Transactional(rollbackFor = Exception.class)  public String confirmSampleMoneyList(HttpServletRequest request,  HttpServletResponse response, ModelMap model) {  String actorId = FinanceServiceImpl.ACTOR\_FINANCE\_MANAGER;  List<Map<String, Object>> list = financeService  .getConfirmSampleMoneyList(actorId);  model.addAttribute("list", list);  model.addAttribute("taskName", "确认样衣制作金");  model.addAttribute("url", "/finance/confirmSampleMoneyDetail.do");  return "/finance/confirmSampleMoneyList";  } |

图4.6 FinanceController的confirmSampleMoneyList方法表

|  |
| --- |
| public List<Map<String, Object>> getOrderList(String actorId,  String taskName) {  List<TaskSummary> tasks = jbpmAPIUtil.getAssignedTasksByTaskname(  actorId, taskName);  List<Map<String, Object>> list = new ArrayList<>();  for (TaskSummary task : tasks) {  Integer orderId = (Integer) jbpmAPIUtil  .getVariable(task, "orderId");  Map<String, Object> model = new HashMap<String, Object>();  Order order = orderDAO.findById(orderId);  model.put("order", order);  model.put("employee", employeeDAO.findById(order.getEmployeeId()));  model.put("task", task);  model.put("taskTime", getTaskTime(task.getCreatedOn()));  model.put("orderId", getOrderId(order));  list.add(model);  }  return list;  } |

图4.7 ServiceUtil的getOrderList方法表

如图4.8所示为点击详情后的订单详情标签页，一共有基本信息、面辅信息、样衣信息、加工信息、版型信息、报价信息和制版费用七个标签页，默认显示制版费用标签页，财务主管也可以选择其他标签页查看具体订单信息。

当财务主管确定制版费用已经收取到之后，将收款信息填入其中，需要填写的信息包括：汇款人、汇款卡号、汇款银行、汇款金额、收款时间、收款账号和备注信息，其中除备注信息为可选之外其他都是必填字段，在其他字段未填写时网页会有提示信息。

当信息填写完整时点击已确认收款按钮完成信息提交，当客户长时间不进行汇款时，财务主管可以选择直接点击未收到汇款按钮结束此订单。汇款信息提交之后，后台使用封装好的jBPM API结束相应的任务，流程流转到下一环节。



图4.8 财务主管样衣金确认任务详情图

### 4.3.2首定金确认

如图4.9所示，该页面为订单首定金确认信息填写页面，首定金确认任务和样衣金确认任务基本相同，只是确认的款项内容不同，在此不做赘述。



图4.9 财务主管首定金确认任务详情图

### 4.3.3尾款金确认

尾款金确认任务和首定金确认任务基本相同，只是确认的款项内容不同，在此不做赘述。

### 4.3.4样衣收取

如图4.10所示，左侧为物流主管的导航菜单，有四个任务，当选择样衣收取后，右侧显示任务列表，有简要信息，点击详情链接后显示订单的详细标签页。



图4.10 物流主管样衣收取任务列表图

如图4.11所示为点击详情后的订单详情标签页，一共有基本信息、面辅信息、样衣信息、加工信息、版型信息五个标签页，默认显示样衣信息标签页，物流主管也可以选择其他标签页查看具体订单信息。

当物流主管收取到样衣之后，点击已收到样衣按钮提交任务，当客户长时间不寄送样衣时，物流主管可以选择直接点击未收到样衣按钮结束此订单。



图4.11 物流主管样衣收取任务详情图

### 4.3.5样衣发货



图4.12 物流主管样衣发货任务详情图

物流主管选择左侧导航菜单中的样衣发货后显示相应任务，之后选择具体订单点击详情，页面跳转至订单详情页面如图4.12所示，物流主管在将样衣发货完成之后将发货信息填写，包括邮寄时间、快递名称和快递单号，最后点击发货完成按钮提交信息。

### 4.3.6产品入库

如图4.13所示，左侧为物流主管的导航菜单，有四个任务，当选择产品入库后，右侧显示任务列表，有简要信息，点击装箱链接后显示订单的详细标签页。



图4.13 物流主管产品入库任务列表图

如图4.14所示，填入装箱信息：依次输入颜色、大小、件数点击添加链接，完成添加，点击删除链接删除已经添加的信息，当一箱的信息添加完成之后点击添加箱号按钮，当所有箱子的装箱信息都添加完成之后点击完成装箱提交信息。



图4.14 物流主管装箱任务详情图

装箱信息填写好之后页面自动跳转到如图4.15界面，依次点击每箱后面的打印链接，如图4.16，打印每个箱子的装包单，依次贴在对应箱子上。



图4.15 物流主管装箱打印任务详情图

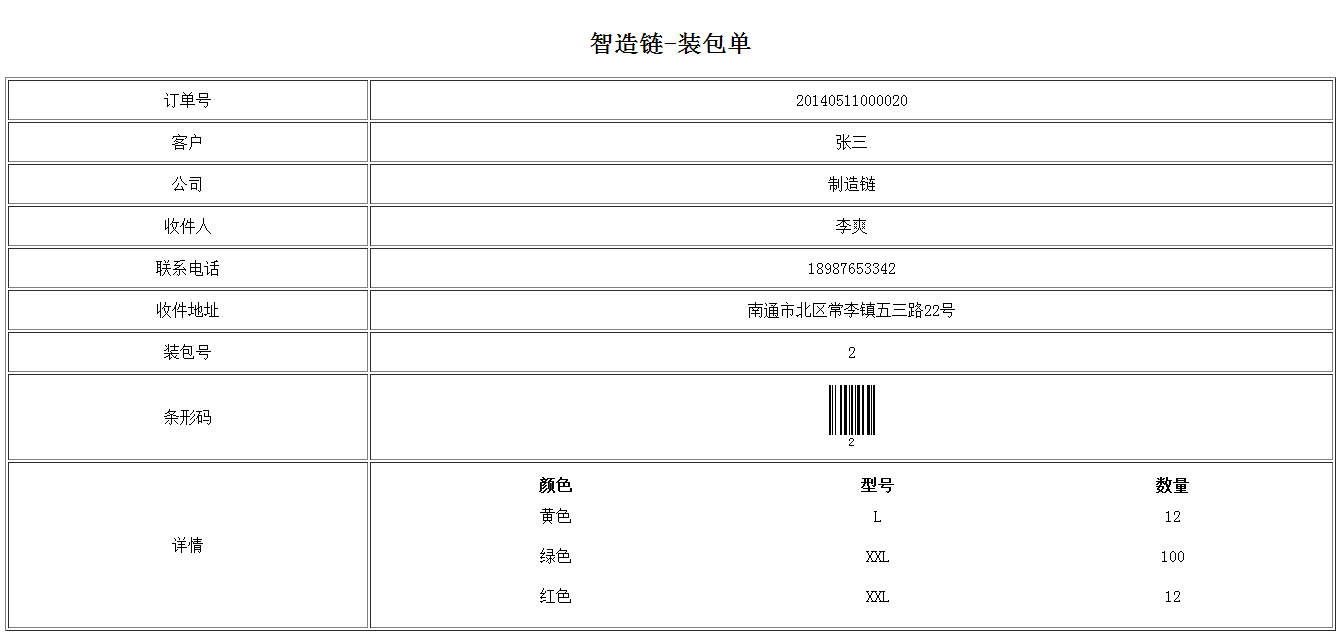


图4.16 物流主管装包单图

我们使用jQuery提供的barcode插件生成条形码，只需在网页中简单调用如下代码[14]，如图4.17所示。

|  |
| --- |
| $(function() {  var pid = $("#pid").val().trim();  $("#bar").barcode(pid, "code128");  }); |

图4.17 条形码生产方法表

之后使用手持扫描设备登陆网站，点击如图4.18所示的入库登记按钮之后选择如图4.19的相应订单。



图4.18 物流主管扫描设备任务图

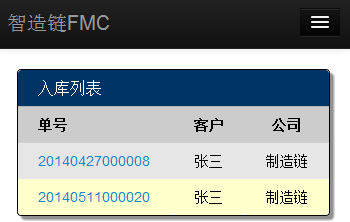


图4.19 物流主管扫描入库任务列表图

之后使用手持设备扫描相应箱子上的条形码，如图4.21所示，在扫描正确的包号之后下面三格（仓库、货架、位置）变为可填写状态，之后填写相关的仓库位置信息点击提交添加信息，全部扫描填写完之后点击完成入库，至此入库完成全部操作。

扫描的动作相当于在input标签中输入了条形码数据并且点击了回车按钮，所以我们需要做的就是监听input标签中的按键事件，之后检测输入的数据，如果包号正确，则将输入框变为可输入状态。具体实现代码如图4.20所示。

|  |
| --- |
| $("#txtScan").keydown(function(e) {  if (e.keyCode === 13) {  checkPackageId($(this).val().trim());  }  }).focus(); |

图4.20 扫描监听方法表



图4.21 物流主管扫描入库任务详情图

### 4.3.7产品发货

手持扫描设备登陆网站，如图4.22所示，点击扫描发货按钮后选择如图4.23所示的相应订单。



图4.22 物流主管扫描设备任务图

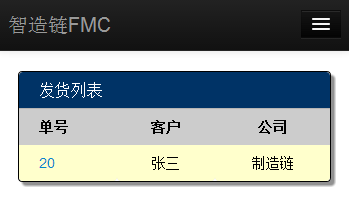


图4.23 物流主管发货扫描任务列表图

紧接着依次扫描所有箱子的条形码，当全部扫描完成之后点击完成扫描按钮，接着线下完成发货。如图4.24所示。



图4.24 物流主管发货扫描任务详情图

如图4.25所示，线下完成发货之后，物流主管使用电脑登陆网页，点击产品发货菜单，选择对应订单。如图4.26所示，填写物流发货信息，包括邮寄时间、快递名称、快递单号、快递价格和备注信息，点击完成发货按钮提交信息，至此产品发货任务完成。



图4.25 物流主管产品发货任务列表图



图4.26 物流主管产品发货任务详情图

## 4.4本章小结

第四章主要介绍了制造链系统中财务部和物流部模块的功能概述、详细设计与实现细节，详细细节包括相应的详细类图、系统顺序图以及相关数据表，而实现部分则展示了相应的流程操作过程、界面截图以及部分代码实现细节。

# 第五章 总结与展望

## 5.1 总结

到目前为止，智造链项目组已经完成了人事部、市场部、设计部、采购部、生产部、质检部、财务部、物流部八个部门的基本功能，同时也完成了客户管理和订单管理的部门。整个服装供应链的具体细节都已经实现，线上功能和线下工作可以很好的结合在一起。智造链系统的第一版本已经完成。

其中，由我完成了财务部、物流部模块的实现，以及整个系统流程管理部分的设计。

整个项目在开发人员和公司工作人员的测试下，修改完善了大部分BUG，基本已经可以上线使用。

## 5.2 展望

智造链系统目前的基本功能已经完善，经过测试之后基本可以投入正式使用。但是，我们依然还能看到其中还有很大改进提升的空间。

一方面，在系统人机交互方面，可以增加更好的用户引导和提示信息，使得用户能够更清晰的了解应该做什么以及能有更加简单人性化的操作

另外一方面，系统的安全性也是不容忽视的问题，作为服装供应链企业，客户的信息对一个企业是十分重要的，防止客户信息的泄露是这个系统应该有的功能，在未来的版本中应该对此做加强和测试。

在未来版本的开发中，将为管理人员提供相应的权限，监测全部的订单流程，从而协调整个供应链的流程进展。同时提供报表功能给企业的业务分析做基础。

除此之外，系统将对外开放，为客户提供一个微信公共账号，使顾客能够及时了解订单的状态信息。相关人员还希望能够提供公告功能以及在线文档功能。

同时，提高系统的可维护性以及性能是我们一直需要改进的地方，还有项目的整体结构可以做一次重构，使得其更加清晰健壮，相信该系统在未来还有很大的塑造空间。

# 参考文献

[1] 百度百科：工作流，http://baike.baidu.com/view/60285.htm?fr=aladdin

[2] Krasimira P. Stoilova and Todor A. Stoilov，Evolution of the workflow management systems，2006

[3] Workflow Management Coalition Terminology & Glossary，1999，7-24页

[4] 维基百科：工作流技术，http://zh.wikipedia.org/wiki/工作流技术

[5] Gary J Nutt，The evolution towards flexible workflow systems，1996

[6] John Koenig，JBoss jBPM White Paper，2004.11

[7] Craig Walls著，张卫滨译，Spring in Action（第三版），北京：人民邮电出版社，2013.06，1-28页

[8] Spring 框架简介，<http://www.ibm.com/developerworks/cn/java/wa-spring1/>

[9] 孙卫琴，精通Hibernate：Java对象持久化技术详解，北京：电子工业出版社， 2005.05，1-44页

[10] jbpm5.3用户手册，<http://docs.jboss.org/jbpm/v5.3/userguide/>

[11] Mauricio Salatino，JBPM 5 Developer Guide，Packt Publishing Limited，2012.12.31，1-30页

[12] 蒋鑫，git权威指南，北京：机械工业出版社，2011.06.30，1-11页

[13] 单东林 张晓菲 魏然，锋利的jQuery（第2版），北京：人民邮电出版社，2012.07.01，1-5页

[14]条形码插件，<http://barcode-coder.com/en/barcode-jquery-plugin-201.html>

# 致谢

整个智造链项目经过两个多月的开发终于完成，随着论文的进一步完善，终于即将完稿。在这期间，虽然在技术上遇到了很多难题，但是最终在同学和老师的帮助下，以及凭借着自己的钻研，都将其一一解决。

首先在这里，我要感谢我的毕业设计指导老师，葛季栋老师。葛老师非常热情，在这两个月的时间里一直很关心我们的项目进度，在我们遇到各种困难的时候，都会主动帮我们寻找各类资源解决问题。

其次，我也要感谢和我一起在项目组中奋斗的同学：王立铭、孙克维和王健，在项目开发期间，他们让我感受到了团队合作的力量，帮助我解决了很多项目中遇到的疑问。

同时也要感谢智造链公司的何越和季老师，在和他们的交流中学习到了实际企业项目中的一些知识，为我们以后迈上社会提供了良好的台阶。

最后我要感谢我的家人和朋友，是他们的关心和支持，使我在大学四年的时光中收获了许多，让我能更加勇敢的向着明天努力奋斗。

在这里祝所有帮助过我和关心我的人一切都好，生活幸福。