

饭店点菜系统的设计与实现

摘 要

近年来，随着人民的生活水平的不断提高，餐饮业的消费持续增长，竞争愈来愈激烈。然而，传统餐饮企业的日常运作还是靠人工管理，从原材料入库到客人点单，再到结账基本上由人工完成记录，这样做不仅耗费人力资源而且容易导致记录丢失或重复等错误，造成管理水平低下。传统的餐饮企业存在这些问题，客人点单需服务员记录并送至厨房，客人结账由手工记录，人力耗费大，客人等待时间长，管理效率低下，这就迫切需要标准的、高效率的计算机管理方式引导其发展，通过计算机管理饭店的日常运作，提高工作效率。

通过饭店点菜系统的开发以实现对餐饮行业提供信息化管理。着重论述了系统功能与实现、数据流程及存储，包括菜单管理、用户管理、订单管理、后台数据库管理等。

运用 JSP 技术和 MySQL 数据库原理，基于 B/S 模式开发了一个饭店点菜系统。本系统在设计中严格按软件工程的理论和方法进一步分析确定软件的模块、数据结构和系统功能的具体实现方案，采取 Windows Vista 为操作系统平台，后台数据库采用 MySQL。

本文阐述了饭店点菜系统的设计与实现过程。运用 HTML 语言，JavaScript, Struts2 与后台数据库链接等关键技术建设饭店点菜系统。

关键词：Struts2，B/S，MySQL，JSP，HTML

Design and Implementation of System for Restaurant Ordering Food

Abstract

In recent years, with the continuous development of living standards, catering consumption continues to grow and competition increase. However, traditional catering businesses still rely on manual management of daily operations, from raw material storage to customers a single point, then check out basically completed by the manual records, this is not only expensive but also prone to human resources records of errors such as missing or repeated, causing management low level. Traditional catering business problem, and customers need a single point of record and sent to the kitchen crew, the guests check out records by hand, a large human cost, the guests wait for a long time, inefficient management, which urgently need a standard, efficient computer management approach to guide its development, through the daily operations of the hotel computer management to improve efficiency.

According to the System for Restaurant Ordering Food, it focuses on the system functionality and implementation, data flow and storage, including menu management, user management, order Management, background database management.

The system is used JSP technology and MySQL database theory which is based on B/S model. It is designed strictly according to the software engineering theory and methods for further analysis to determine the software modules, data structures and systems function in the program, to take Windows Vista for the operating system platform, the background database using MySQL. This sub-section the following a la carte system, described the System for Restaurant Ordering Food design and implementation.

This article described the restaurant ordering food system design and implementation. Use of HTML language, Freemarker technology, JavaScript, Spring, Struts2 links with key technologies such as the background database building the System for Restaurant Ordering Food.

Key words: Struts2 , B/S , MySQL , JSP , HTML

目录

第 1 章 绪 论 1

1.1 课题研究背景1..

1.2 课题研究目的1..

1.3 课题研究意义1..

1.4 国内外现状1..

第 2 章 关键技术介绍 2

2.1 STRUTS22..

2.2 MYSQL2..

2.3 TOMCAT2..

2.4 MYECLIPSE2..

第 3 章 系统分析 4

3.1 构架概述4..

3.1.1 功能构架 4

3.1.2 模块需求描述 4

3.2 系统开发环境5..

3.3 系统任务的可行性分析5..

3.3.1 软件可行性分析错误！未定义书签。

3.3.2 技术可行性分析错误！未定义书签。

3.3.3 系统安全性分析错误！未定义书签。

第 4 章 系统设计 7

4.1 设计指导思想和原则 7

4.1.1 指导思想 7

4.1.2 软件设计原则 7

4.2 构架概述7..

4.3 系统的功能结构设计7..

4.3.1 模块需求概要描述 7

4.3.2 模块需求详细描述 8

4.4 系统控制流程8..

4.5 数据字典9..

第 5 章 系统实现 11

5.1 环境配置 11

5.1.1 JDK1.6 配置 11

5.1.2 修改 Tomcat 默认配置 11

5.1.3 项目部署 11

5.2 用户登陆功能实现 11

5.3 界面设计12

5.3.1 系统登录界面 12

5.3.2 用户浏览菜肴界面 12

5.3.3 普通用户注册界面 13

5.3.4 用户确认提交订单界面 13

5.3.5 后台管理界面 14

5.3.6 后台查看用户订单界面 14

第 6 章 系统测试 16

6.1 用户登陆和注册功能测试16

6.2 用户点菜功能测试16

6.3 订单功能测试16

6.4 用户订单提交功能测试16

6.5 用户添加菜肴到订单功能测试16

6.6 管理员菜肴管理功能测试16

6.7 管理员订单管理功能测试16

6.8 管理员用户管理模块测试17

第 7 章 结 论 18

参考文献 19

致 谢 20

第 1 章 绪 论

近年来，随着居民消费水平的提高，居民外出就餐的次数增多，消费增加，大众化成为目前我国餐饮市场的主流。然而，目前餐饮业结构不尽合理，中高档餐馆发展较快、数量过剩，需求不足，而大众需求较大的中低档餐馆普遍存在着脏、乱、差的现象，处于低水平的发展阶段，难以满足广大消费者的需求。这种情况下，饭店业提出了饭店餐饮大众化经营，引起了业内人士的关注。

所谓饭店餐饮大众化经营是指以饭店自身的硬件和软件优势为依托，以大众化的原料、高超的厨艺、较低的价格向社会大众提供较高质量标准的餐饮产品、良好的环境和相应服务，从而使饭店成为社会绝大部分消费者服务的经营方式。即饭店将广大平民百姓作为其目标市场，以他们所能接受的价格为其提供餐饮品种和相应服务。这里所说的价格并不是指绝对价格无限制的低，而是指相对于其出品质量水平和服务标准来说，价格较低。

1.1 课题研究背景

在饭店的日常经营中，传统的点菜方式对客人和服务员来说，都会带来很多没必要的麻烦，本系统提供方便快捷的点菜服务，可以客人在无需服务员的情况下完成饭菜的查询、点单等操作，极大的方便了顾客的点菜，而且在满足顾客需求的同时节省了相应的人力、物力资源。

1.2 课题研究目的

为克服传统点菜方式的弊端，设计采用电子点菜系统，客人点菜的时候，基本不需要服务员的陪伴，就能实时地了解菜的特色，且对已点过的菜的数量和价钱一目了然，方便了顾客的消费，同时也减轻了餐馆的服务压力。在餐饮业得以推广，可大大提高服务质量，并可节省许多人力物力。

作为计算机应用的一部分，使用计算机对餐饮服务信息进行管理，具有手工管理所无法比拟的优点。例如：检索迅速、查找方便、可靠性高、存储量大、保密性好、寿命长、成本低等。这些优点能够极大地提高信息管理的效率，也是企业的科学化、正规化管理与世界接轨的重要条件。

1.3 课题研究意义

本饭店管理系统广泛适用于中小型餐饮行业，为方便饭店管理人员更有效的管理饭店订单而开发，该系统可以帮助餐饮行业处理日常的点菜业务，规范管理工作和操作流程，是餐饮行业的得力助手。

本系统如在餐饮业得以推广，方便了顾客的消费，同时也减轻了餐馆的服务压力。大大提高服务质量，并可节省大量人力、物力资源，并提高饭店的工作效率。可以帮助饭店在餐饮业的竞争中取得先机。

1.4 国内外现状

近年来，国内出现的点菜系统多为掌上电脑 PDA 无线点菜，这种系统就需要针对每个饭店，来进行订制开发，成本高，同时不便于管理。顾客不能直观的看到自己所点的菜单，以及价目。在点菜时，顾客还要通过翻阅纸质的菜单进行点菜，反而加大了成本，造成诸多不便。

第 2 章 关键技术介绍

2.1 Struts2

Struts2 是第二代基于 Model-View-Controller(MVC) 模型的 Web 应用框架。Struts2 是 Java 企业级 Web 应用的可扩展性的框架。它是 WebWork 和 Struts 社区合并后的产物。这一版本的 Struts2 声称，Struts2 会接近于原先版本 Struts，并且会更容易使用。Struts 2 没有继承 Struts1 的血统，而是继承了 Webwork 的血统。

Struts2 提供了对 MVC 的一个清晰的实现，这一实现包含了很多参与对所以请求进行处理的关键组件，如：拦截器、OGNL 表达式语言、堆栈。

虽然 Struts2 号称是一个全新的 框架，但这仅仅是相对 Struts1 而言。Struts2 与 Struts1 相比，确实有很多革命性的改进，但它并不是新发布的新框架，而是在另一个赫赫有名的框架：WebWork 基础上发展起来的。从某种程度上来讲，Struts2 没有继承 Struts 1 的血统，而是继承 WebWork 的血统。或者说，WebWork 衍生出了 Struts2，而不是 Struts 1 衍生了 Struts2。因为 Struts2 是 WebWork 的升级，而不是一个全新的框架，因此稳定性、性能等各方面都有很好的保证：而且吸收了 Struts1 和 WebWork 两者的优势，因此，是一个非常值得期待的框架。

Apache Struts2 是一个优雅的，可扩展的 JAVAEE web 框架。框架设计的目标贯穿整个开发周期，从开发到发布，包括维护的整个过程。

Apache Struts2 即是之前大家所熟知的 WebWork2。在经历了几年的各自发展后，WebWork 和 Struts 社区决定合二为一，也即是 Struts2。

2.2 MySQL

MySQL 最初的开发者的意图是用 MySQL 和他们自己的快速低级例程 (ISAM) 去连接表格。不管怎样，在经过一些测试后，开发者得出结论：MySQL 并没有他们需要的那么快和灵活。这导致了一个使用几乎和 MySQL 一样的 API 接口的用于他们的数据库的新的 SQL 接口的产生，这样，这个 API 被设计成允许为用于 MySQL 而写的第三方代码更容易移植到 MySQL。

2.3 Tomcat

Tomcat 很受广大程序员的喜欢，因为其运行时占用的系统资源小，扩展性好，支持负载均衡与邮件服务等开发应用系统常用的功能；而且还在不断的改进和完善中，任何一个感兴趣的程序员都可以更改它或在其中加入新的功能。

Tomcat 是一个小型的轻量级应用服务器，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试 JSP 程序的首选。对于一个初学者来说，可以这样认为，当在一台机器上配置好 Apache 服务器，可利用响应对 HTML 页面的访问请求。实际上 Tomcat 部分是 Apache 服务器的扩展，但是独立运行的，所以当运行 Tomcat 时，实际上作为一个与 Apache 独立的进程单独运行的。

这里的诀窍是，当配置正确时，Apache 为 HTML 页面服务，而 Tomcat 实际上运行 JSP 页面和 Servlet。另外，Tomcat 和 IIS、Apache 等 Web 服务器一样，具有处理 HTML 页面的功能，另外它还是一个 Servlet 和 JSP 容器，独立的 Servlet 容器是 Tomcat 的默认模式。不过，Tomcat 处理静态 HTML 的能力不如 Apache 服务器。

2.4 MyEclipse

MyEclipse 企业级工作平台 (MyEclipse Enterprise Workbench, 简称 MyEclipse) 是对 Eclipse IDE 的扩展，可以在数据库和 JavaEE 的开发、发布，以及应用程序服务器的整合

方面极大的提高工作效率。功能丰富的 JavaEE 集成开发环境，包括了完备的编码、调试、测试和发布功能，完整支持 HTML, Struts, JSF, CSS, Javascript, SQL, Hibernate

在结构上，MyEclipse 的特征可以被分为 7 类：

- (1) JavaEE 模型；
- (2) WEB 开发工具；
- (3) EJB 开发工具；
- (4) 应用程序服务器的连接器；
- (5) JavaEE 项目部署服务；
- (6) 数据库服务；
- (7) MyEclipse 整合帮助。

对于以上每一种功能上的类别，在 Eclipse 中都有相应的功能部件，并通过一系列的插件来实现。MyEclipse 结构上的这种模块化，可以让人们在不影响其他模块的情况下，对任意模块进行单独的扩展和升级。

简单而言，MyEclipse 是 Eclipse 的插件，也是一款功能强大的 JavaEE 集成开发环境，支持代码编写、配置、测试以及除错，MyEclipse6.0 以前版本需先安装 Eclipse。MyEclipse6.0 以后版本安装时不需安装 Eclipse。

第 3 章 系统分析

3.1 构架概述

3.1.1 功能构架

根据对饭店这个工作流程需求调研结果，确定本系统前台普通用户操作界面包括以下功能，如图 3.1 所示。

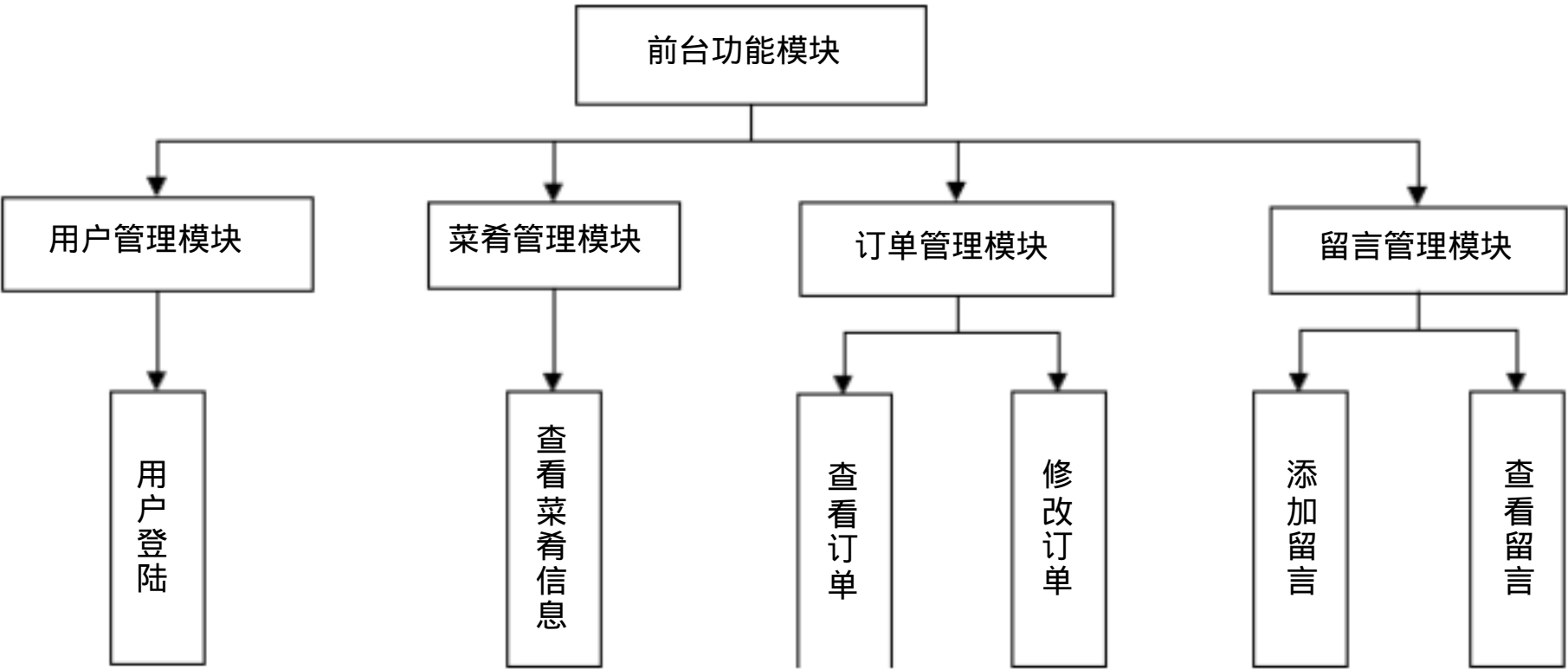


图 3.1 前台用户功能模块图

根据对饭店这个工作流程需求调研结果，确定本系统管理员操作界面主要包括以下功能，如图 3.2 所示。

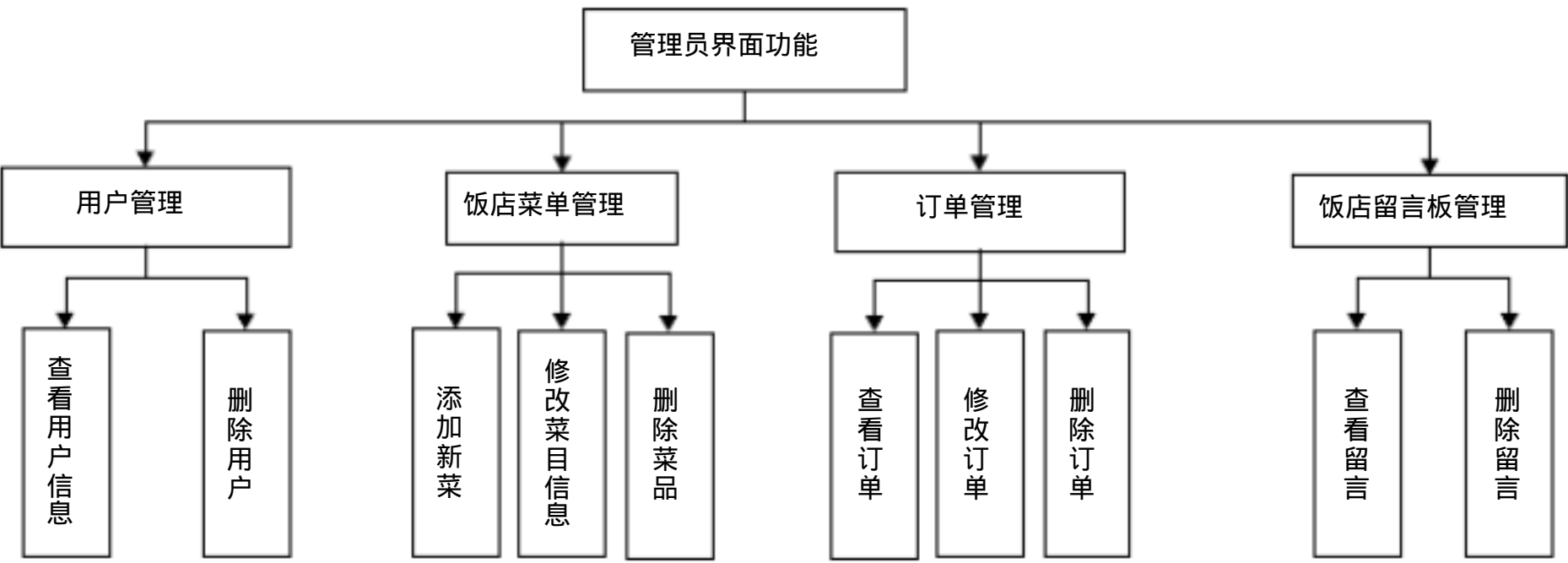


图 3.2 管理员功能模块图

图 3.2 为管理员管理界面所有的功能。

3.1.2 模块需求描述

系统功能模块分为用户管理模块、菜肴管理模块、订单管理模块、留言管理模块四部分，如图 3.3 所示

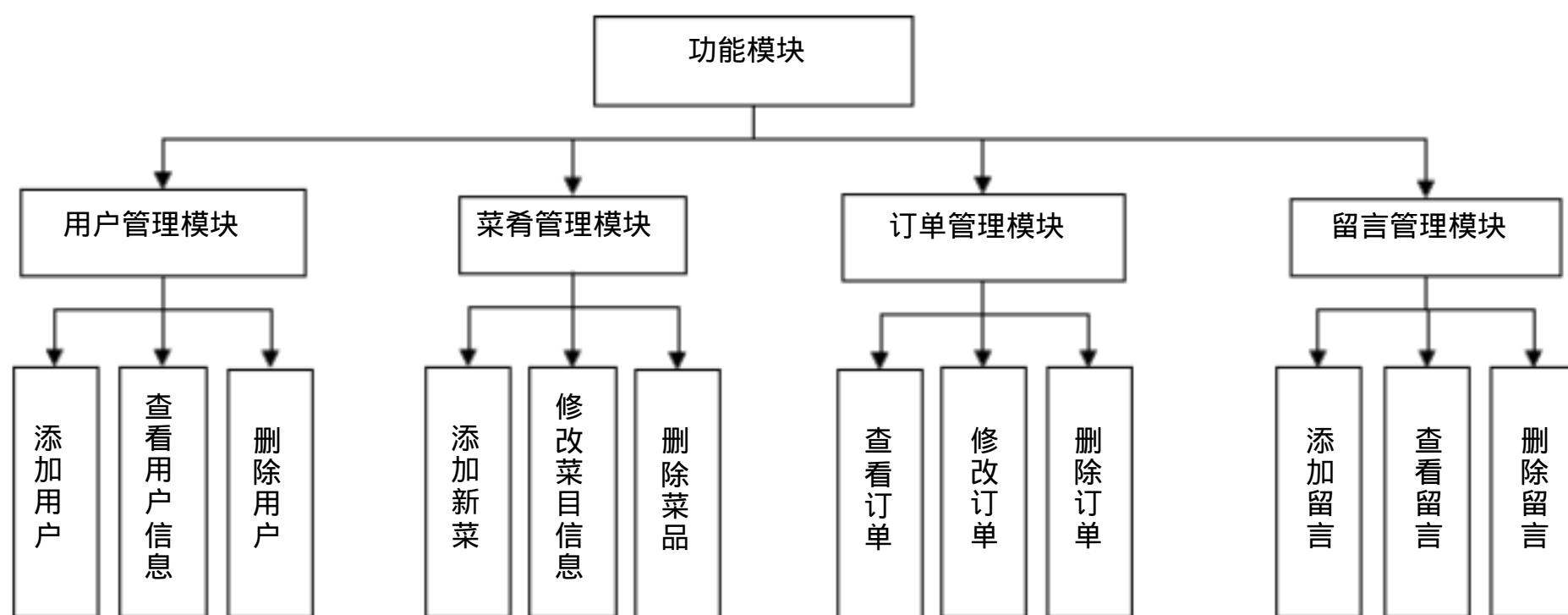


图 3.3 系统功能模块图

（1）用户管理模块

对系统中用户进行统一管理，注册的新用户只拥有普通用户的权限，管理员用户在后台查看所有用户信息，删除用户。不能对用户进行修改和删除。

（2）订单管理模块

订单模块主要记录了每个用户提交的信息，每个用户的订单都是唯一的，可同时存在多个用户订单，订单模块根据不同的用户等级和所对应的唯一用户，拥有不同的权限。管理员账户可以查看、修改删除所有用户的订单，普通用户可以查看订单，修改未提交的订单。

（3）菜单管理模块

管理整个系统中的菜肴，管理员账户通过后台管理，可以添加新的菜肴，删除现有的菜肴，修改已有菜肴的价格，介绍等信息。

（4）留言管理模块

详细、清晰、规范的留言功能，此功能可以让客人指出饭店所存在的不足，改善饭店的服务水平，普通用户都可以登陆系统添加新的留言记录，管理员用户进入后台对留言进行管理，可以删除留言。

3.2 系统开发环境

硬件环境：CPU T7500 2.2GHz；内存 2GB；120GB 硬盘；

软件环境：Windows Vista sp2 /Java 6.0;

开发工具：Myeclipse7.5；

数据库服务器：MySQL 5.0；

WEB 服务器：Tomcat 6.0。

3.3 系统任务的可行性分析

3.3.1 软件可行性分析

（1）单机版

可以选择的操作系统包括 Windows xp/2000 及更高版本。

在 Windows 2000 安装和运行要求有 administrative 权限。

数据库选择使用 MySQL 5.0。

（2）网络版

服务器端可选用 Windows Server 2003 客户端可选择 Windows xp。

服务器端需要安装 MySQL 数据库。

本系统属于数据库应用程序，本系统需要一个数据库服务器及其运行的操作系统平

台，根据客户的数据量并不是十分巨大，选择目前市场上价格比较低廉的数据库服务器产品：MySQL 5.0，操作系统平台则客户机端可以选择 Windows xp。服务器端可以选择 Windows Server 2003。

3.3.2 技术可行性分析

前台开发工具采用 JSP,利用 Web 页面的易浏览操作的优点，能快速开发出应用页面。这些软件在中小型 MIS 开发中已被大量应用，技术上都比较成熟，因此技术上是可行的。控制层用 Struts2 进行控制转发，以 Struts2 为核心架构，基于 MVC 模式的指导思想，以 Struts2 的 Action 处理业务逻辑。

3.3.3 系统安全性分析

系统采用 MySQL 数据库，保证了信息系统的安全可靠。系统登录方式采用用户名和密码结合方式登陆。系统操作员采用严格的权限设置，每个用户具有独立的权限。以 Windows Server 2003 作为网络运行环境，采用客户 / 服务器 (Client-Server)，任何操作变化会时时反映到服务器端。

第 4 章 系统设计

4.1 设计指导思想和原则

4.1.1 指导思想

- (1) 在饭店的日常经营中，传统的点菜方式对客人和服务员来说，都会带来很多不必要的麻烦，本系统提供方便快捷的点菜服务，客人可以在无需服务员的情况下完成饭菜的查询、点单等操作，极大的方便了顾客的点菜，而且在满足顾客需求的同时节省了相应的人力、物力资源。
- (2) 实施该系统也能给饭店带来长期效益，帮助饭店实现管理方法的现代化、科学化，极大地提高饭店的工作质量与工作效率，减少管理支出的费用，从而保证饭店管理的可持续的良好发展。
- (3) 本饭店点菜系统对用户的教育水平与技术专长要求不高，一般具备电脑操作基本知识就可以完成系统的操作。

4.1.2 软件设计原则

- (1) 系统可以长期运行，版本也可以进行不断升级，来满足用户的新需求。
- (2) 系统是用 Myeclipse7.5 开发的，大部分的操作系统都能顺利运行， 适合餐饮行业使用。同时不需要专门的人员来维护系统以及系统的数据库。
- (3) 系统操作简单，饭店的工作人员只需掌握一些基本的电脑使用方面的知识，简单进行培训就能够轻松使用本系统，无需饭店花费大量财力物力来培训员工。
- (4) 系统一经投入便能够运行，并产生效益得到回报。
- (5) 在控制层采用 Struts2 技术时，在相应的 action 类中，创建相应的访问器。

4.2 构架概述

本系统不仅严格按 MVC 模式设计，还按 J2EE 分层设计，将中间层严格分成业务逻辑层、DAO 层及数据持久层等。 MVC 层的控制器绝对禁止持久层访问，甚至不参与业务逻辑的实现。表现层采用传统 JSP 技术。

本系统采用的是典型的 J2EE 三层结构，分为表现层、中间层 (业务逻辑层)和数据服务层。三层体系将业务规则、数据访问及合法性校验等工作放在中间层处理。客户端不直接与数据库交互，而是通过组件与中间层建立连接，再由中间层与数据库交互。 JSP 广泛的应用和稳定的表现，为其作为表现层技术打下了坚实的基础。为了将控制层与业务逻辑层分离，又细分为以下几种。

Web 层，就是 MVC 模式里面的 C，负责逻辑层和表现层的交互。调用业务逻辑层，并将业务数据返回给表现层作组织表现，该系统的 MVC 采用 Struts 框架。

Service 层(就是业务逻辑层)，负责实现业务逻辑。业务逻辑层以 DAO 层为基础，通过对 DAO 组件的正面模式包装，完成系统所要求的业务逻辑。

DAO 层，负责与持久化对象交互。该层封装了数据的增、删、查、改的操作。

PO，持久化对象。通过实体关系映射工具将关系型数据库的数据映射成对象，很方便地实现以面向对象方式操作数据库，其数据服务层用来存放数据。

4.3 系统的功能结构设计

4.3.1 模块需求概要描述

模块需求概要描述，如表 4.1 所示。

表 4.1 模块需求概要描述

功能编号	功能名称	功能描述
1	用户管理	用户模块对系统中用户进行统一管理，注册的新用户只拥有普通用户的权限，管理员用户在后台查看所有用户信息，删除用户。 不能对用户进行修改和删除。
2	菜单信息管理	管理整个系统中的菜肴，管理员账户通过后台管理，可以添加新的菜肴，删除现有的菜肴，修改已有菜肴的价格，介绍等信息。
3	订单管理	订单模块主要记录了每个用户提交的信息，每个用户的订单都是唯一的，可同时存在多个用户订单，订单模块根据不同的用户等级和所对应的唯一用户，拥有不同的权限。管理员账户可以查看、修改删除所有用户的订单，普通用户可以查看订单，修改未提交的订单
4	留言管理	详细、清晰、规范的留言功能，普通用户都可以登陆系统添加新的留言记录，管理员用户进入后台对留言进行管理，可以删除留言。

4.3.2 模块需求详细描述

(1) 用户管理模块

对系统中用户进行统一管理，注册的新用户只拥有普通用户的权限，管理员用户在后台查看所有用户信息，删除用户。不能对用户进行修改和删除。

(2) 订单管理模块

订单模块主要记录了每个用户提交的信息，每个用户的订单都是唯一的，可同时存在多个用户订单，订单模块根据不同的用户等级和所对应的唯一用户，拥有不同的权限。管理员账户可以查看、修改删除所有用户的订单，普通用户可以查看订单，修改未提交的订单。

(3) 菜单管理模块

管理整个系统中的菜肴，管理员账户通过后台管理，可以添加新的菜肴，删除现有的菜肴，修改已有菜肴的价格，介绍等信息。

(4) 留言管理模块

详细、清晰、规范的留言功能，此功能可以让客人指出饭店所存在的不足，改善饭店的服务水平，普通用户都可以登陆系统添加新的留言记录，管理员用户进入后台对留言进行管理，可以删除留言。

4.4 系统控制流程

用户打开系统，先进行登陆认证，然后开始选择要点的菜肴，选择完毕后，用户查看购物车，校对订单，确认无误后提交订单，如需加菜，则再添加菜肴到购物车，提交新的订单，用餐完毕后，进行买单。如图 4.1 所示。

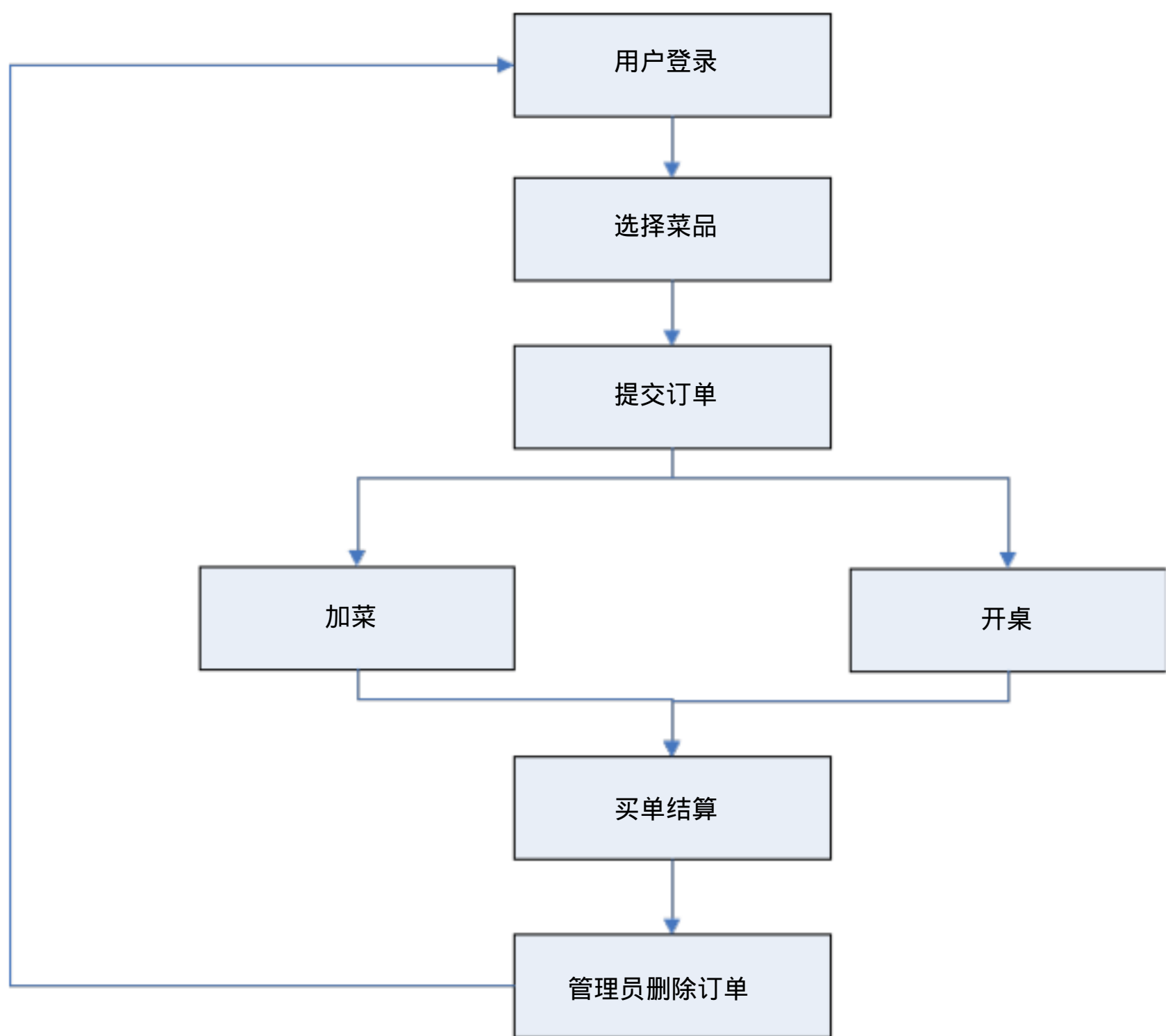


图 4.1 系统控制流程图

图 4.1 主要描述了系统整个控制流程。

4.5 数据字典

(1) 用户表 user table

用户表用于存储用户编号，用户名称，用户密码。如表 4.2 所示。

表 4.2 用户表

字 段 名 称	字 段 描 述	主 键	类 型	长 度	说 明
userid	用户编号	是	Int	11	非空
username	用户名称	不是	Varchar	20	非空
password	用户密码	不是	Varchar	20	非空

(2) 菜肴表 dish table

菜肴表用于定义菜肴编号，菜肴名称，菜肴描述，菜肴价格。如表 4.3 所示。

表 4.3 菜肴表

字 段 名 称	字 段 描 述	主 键	类 型	长 度	说 明
dishid	菜肴编号	是	Int	11	非空
dishname	菜肴名称	不是	Text	0	非空

description	菜肴描述	不是	Text	0	
price	菜肴价格	不是	Int	10	非空

(3) 订单表 order table

订单表用于定义订单编号，用户姓名，菜肴编号。如表 4.4 所示。

表 4.4 订单表

字 段 名 称	字 段 描 述	主 键	类 型	长 度	说 明
mesid	订单编号	是	Int	11	非空
username	用户姓名	不是	Varchar	20	非空
dishid	菜肴编号	不是	Text	11	非空
remark	标记信息	不是	Text	0	

(4) 留言表 message table

留言表用于定义留言编号，用户姓名，留言信息。如表 4.5 所示。

表 4.5 留言表

字 段 名 称	字 段 描 述	主 键	类 型	长 度	说 明
mesid	留言编号	是	Int	11	非空
username	用户姓名	不是	Varchar	20	非空
Content	留言信息	不是	Text	11	非空

第 5 章 系统实现

5.1 环境配置

5.1.1 JDK1.6 配置

在把 jdk1.6 安装到计算机中之后，我们来进行设置使 Java 环境能够使用。

首先右键点我的电脑。打开属性。然后选择“高级”里面的“环境变量”，在新的打开界面中的系统变量需要设置三个属性“JAVA_HOME”、“path”、“classpath”其中在没安装过 jdk 的环境下。path 属性是本来存在的。而 JAVA_HOME 和 classpath 是不存在的。点“新建”，然后在变量名写上 JAVA_HOME，顾名思义该变量的含义就是 Java 的安装路径，然后在变量值写入刚才安装的路径“C:\jdk1.6”。（注：如果安装的路径不是磁盘 C 或者不是在 jdk1.6 这个文件夹，可对应修改。以下文字都是假定安装在 C:\jdk1.6 里面。）其次在系统变量里面找到 path，然后点编辑，path 变量的含义就是系统在任何路径下都可以识别 Java 命令，则变量值为“.;%JAVA_HOME%\bin”，（其中“%JAVA_HOME%”的意思为刚才设置 JAVA_HOME 的值），也可以直接写上“C:\jdk1.6\bin”。

最后再点“新建”，然后在变量名上写 classpath 该变量的含义是为 Java 加载类 (class or lib) 路径，只有类在 classpath 中，Java 命令才能识别。其值为“.;%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar（要加 . 表示当前路径）”，与相同“%JAVA_HOME%”有相同意思。

以上三个变量设置完毕，则按“确定”直至属性窗口消失，下来是验证看看安装是否成功。先打开“开始”->“运行”，打入“cmd”，进入 dos 系统界面。然后打“Javaversion”，如果安装成功。系统会显示 Java version jdk"1.6.0"。

确保安装在 C 盘文件名为 jdk1.6 环境变量直接复制就可以了，在 dos 界面中输入 javac 来查看该命令是否合法，同样输入 Java 来查看该命令是否合法。

5.1.2 修改 Tomcat 默认配置

Apache Tomcat 是一个 Web 容器，允许您为 Java Servlets 和 JSP (Java 服务器页面) 的 Web 应用提供服务。默认情况下，Tomcat 6.0 在 8080 端口上运行一个 HTTP 连接器，为避免与系统中其它服务器发生冲突，您可能需要修改这些默认的端口。可以在 Apache Tomcat 的安装路径，通过改 ./conf/server.xml 中如下行来实现：

```
<Connector port="8080" protocol="HTTP/1.1"
            connectionTimeout="20000"
            redirectPort="8443" />
```

5.1.3 项目部署

将项目导入开发工具 MyEclipse7.5 后，在 Tomcat 安装路径 conf\Catalina\localhost 下新建文件“ShoppingOnline.xml”文件。内容如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<Context docBase="E:\workspace\ShoppingOnline\webapp" path="/ShoppingOnline"
        privileged="true" antiResourceLocking="false" antiJARLocking="false" reloadable="true">
</Context>
```

5.2 用户登陆功能实现

用户登陆时验证是否存在该用户，密码是否正确，内容如下：

```
public class UserService {
    public User login(String userName) {
        Connection conn = DB.createConn();
        String sql = "select * from user where username = ?";
        PreparedStatement ps = DB.prepare(conn, sql);
        User user = null;
```