摘 要

现如今随着教育的不断发展，学生人数也不断增加。学生的信息管理是学校经常遇到的问题。各大高校招生规模的逐步扩大，对学生信息的管理也不断在改革，虽然在校学生不断的增加，但是学生管理人员则相对减少。加上我国高等教育基层学生的信息内容杂，但是管理要求越来越细，传统管理办法已基本不适应新形势的要求。所以在计算机飞速发展的今天，实现学生信息的管理是可行而必要的工作，学生管理信息系统，既能够提高工作效率，又可以提高工作水平。

计算机具有运算速度快、正确、能按照设计逻辑处理问题等特性，在学生信息的录入、统计中如采用一个计算机化的信息系统进行处理，就不会发生信息遗漏或者数据输入不正确的情形。

作为计算机应用的一部分,使用计算机对学生档案进行管理,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的效率,也是学校科学化、正规化管理的重要条件。

关键词：收藏管理； 云收藏； Java Web工程； Spring MVC；

**ABSTRACT**

Nowadays, with the continuous development of education, the number of students is also increasing. Information management of students is a problem that schools often encounter. With the gradual expansion of the enrollment scale of universities, the management of student information is also constantly being reformed. Although the number of students in the school continues to increase, the number of student administrators is relatively reduced. In addition, the information content of grass-roots students in China's higher education is complex, but the management requirements are becoming more and more detailed, and the traditional management methods have basically not adapted to the requirements of the new situation. So in the computer rapid development today, the realization of student information management is feasible and necessary work, student management information system, not only can improve work efficiency, but also can improve the level of work.

The computer has the characteristics of fast operation speed, correct, and can deal with problems according to the design logic. If a computerized information system is used to deal with the input and statistics of students' information, there will be no information omission or incorrect data input.

As a part of the computer application, the use of computer to manage the student archives, with manual management can not be compared with the advantages. For example: search fast, find convenient, high reliability, large storage, good confidentiality, long life, low cost. These advantages can greatly improve the efficiency of the school is also an important condition of scientific and standardized management.

**Key words:** Collection management; Cloud collection; Java Web Engineering; Spring MVC.

目 录

[1绪论 1](#_Toc77192794)

[1.1 研究背景 1](#_Toc77192795)

[1.2 研究目的 1](#_Toc77192796)

[1.3 研究意义 1](#_Toc77192797)

[1.4 研究内容 1](#_Toc77192798)

[1.5 本文结构 2](#_Toc77192799)

[2 相关技术综述 2](#_Toc77192800)

[2.1 java 2](#_Toc77192801)

[2.1.1 java简介 2](#_Toc77192802)

[2.1.2 java特性 3](#_Toc77192803)

[2.2 Spring MVC框架 3](#_Toc77192804)

[2.2.1 Spring MVC框架简介 3](#_Toc77192805)

[2.2.2 Spring MVC框架特性 3](#_Toc77192806)

[2.3 MyBatis框架 3](#_Toc77192807)

[2.3.1 MyBatis框架简介 4](#_Toc77192808)

[2.3.2 MyBatis框架特性 4](#_Toc77192809)

[2.4 开发环境介绍 4](#_Toc77192810)

[2.4.1 Eclipse简介 4](#_Toc77192811)

[2.4.2 MySql简介 5](#_Toc77192812)

[2.4.3 Tomcat8.0简介 5](#_Toc77192813)

[2.4.4 MVC 设计模式 5](#_Toc77192814)

[2.4.5 Spring 框架 6](#_Toc77192815)

[3 需求分析 6](#_Toc77192816)

[3.1 功能需求分析 6](#_Toc77192817)

[4 系统设计 7](#_Toc77192818)

[4.1 软件架构设计 7](#_Toc77192819)

[4.1 数据库实现 7](#_Toc77192820)

[4.2 登录注册功能实现 8](#_Toc77192821)

[4.2.1 登录注册功能实现思路 8](#_Toc77192822)

[4.2.2 登录注册页面功能实现 8](#_Toc77192823)

[4.3 学生管理系统主页面功能实现 10](#_Toc77192824)

[4.3.1 学生管理系统主页面功能实现思路 10](#_Toc77192825)

[4.3.2 主页面功能实现 10](#_Toc77192826)

[4.4 学生信息插入功能实现 11](#_Toc77192827)

[4.4.1 学生管理系统学生信息插入功能实现思路 11](#_Toc77192828)

[4.4.2 学生信息插入页面功能实现 11](#_Toc77192829)

[4.5 学生信息更新功能实现 12](#_Toc77192830)

[4.5.1 学生管理系统学生信息更新功能实现思路 12](#_Toc77192831)

[4.5.2 学生信息更新页面功能实现 12](#_Toc77192832)

[4.6 学生信息刷新功能实现 13](#_Toc77192833)

[4.6.1 学生管理系统学生信息刷新功能实现思路 13](#_Toc77192834)

[4.6.2 学生信息更新页面功能实现 13](#_Toc77192835)

[4.7 学生信息删除功能实现 14](#_Toc77192836)

[4.7.1 学生管理系统学生信息刹车功能实现思路 14](#_Toc77192837)

[4.7.2 学生信息更新页面功能实现 14](#_Toc77192838)

[4.8 学生管理系统主页面功能实现 15](#_Toc77192839)

[4.8.1.1 学生管理系统主页面功能实现思路 15](#_Toc77192840)

[4.8.1.2 主页面功能实现 16](#_Toc77192841)

[5 系统实现 16](#_Toc77192842)

[5.1 开发环境搭建 16](#_Toc77192843)

[5.1.1 java环境 16](#_Toc77192844)

[5.1.2 MyEclipse10客户端环境安装 17](#_Toc77192845)

[5.1.3 MySQL数据库环境搭建 17](#_Toc77192846)

[5.1.4 Navicat客户端环境搭建 18](#_Toc77192847)

[5.1.5 Tomcat7.0客户端环境搭建 18](#_Toc77192848)

[5.1.6 Spring MVC框架环境配置搭建 19](#_Toc77192849)

[5.1.7 MyBatis框架环境配置搭建 20](#_Toc77192850)

[5.1.8 阿里云服务器环境搭建 21](#_Toc77192851)

[5.2 项目结构设计 24](#_Toc77192852)

[5.3 用户登录注册子系统 26](#_Toc77192853)

[5.3.1 登录功能 26](#_Toc77192854)

[5.3.2 注册功能 26](#_Toc77192855)

[5.4 用户收藏网页子系统 27](#_Toc77192856)

[5.4.1 链接收藏网页功能 27](#_Toc77192857)

[5.4.2 网页快照收藏功能 28](#_Toc77192858)

[5.4.3 生成新网页收藏功能 28](#_Toc77192859)

[5.5 用户管理收藏网页子系统 29](#_Toc77192860)

[5.5.1 用户查看收藏的某网页功能 29](#_Toc77192861)

[5.5.2 用户修改收藏的某网页功能 29](#_Toc77192862)

[5.5.3 用户删除收藏的某网页功能 30](#_Toc77192863)

[5.6 网页分享社区子系统 30](#_Toc77192864)

[5.6.1 分享网页到平台社区功能 30](#_Toc77192865)

[5.6.2 新增平台好友功能 31](#_Toc77192866)

[5.6.3 平台好友聊天功能 31](#_Toc77192867)

[5.6.4 分享网页给好友功能 32](#_Toc77192868)

[5.6.5 收藏平台中他人或好友分享的网页功能 32](#_Toc77192869)

[5.6.6 取消自己分享的网页功能 33](#_Toc77192870)

[5.6.7 查看他人或好友分享的网页功能 33](#_Toc77192871)

[5.7 后台管理子系统 33](#_Toc77192872)

[5.7.1 浏览用户网页收藏信息功能 33](#_Toc77192873)

[5.7.2 删除用户网页收藏信息功能 34](#_Toc77192874)

[5.7.3 删除某用户功能 34](#_Toc77192875)

[6 结束语 35](#_Toc77192876)

[参考文献 36](#_Toc77192877)

[附 录 37](#_Toc77192878)

[附录1：登录功能代码 37](#_Toc77192879)

[附录2：注册功能代码 38](#_Toc77192880)

[附录3：网页快照收藏功能代码 39](#_Toc77192881)

[附录4：生成新网页收藏功能代码 40](#_Toc77192882)

[附录5：用户修改收藏的某网页功能代码 42](#_Toc77192883)

[附录6：用户删除收藏的某网页功能代码 42](#_Toc77192884)

[附录7：分享网页到平台社区功能代码 43](#_Toc77192885)

[附录8：新增平台好友功能代码 43](#_Toc77192886)

[附录9：平台好友聊天功能代码 45](#_Toc77192887)

[附录10：分享网页给好友功能代码 47](#_Toc77192888)

[附录11：收藏平台中他人或好友分享的网页功能代码 47](#_Toc77192889)

[附录12：取消自己分享的网页功能代码 48](#_Toc77192890)

[附录13：浏览用户网页收藏信息功能代码 48](#_Toc77192891)

[附录14：删除某用户功能代码 50](#_Toc77192892)

[致 谢 52](#_Toc77192893)

1绪论

1.1 研究背景

学生信息管理系统是一个教育单位不可缺少的部分。一个功能齐全、简单易用的信息管理系统不但能有效地减轻学校相关工作人员的工作负担,它的内容对于学校的决策者和管理者来说都至关重要。所以学生信息管理系统应该能够为用户提供充足的信息和快捷的查询手段。设计一个学生基本信息管理系统，实现“系统登录”，“学生信息增加”，“学生信息查询”，“学生信息删除”，“学生信息修改”，“学生信息导入”，“学生信息导出”等功能。

1.2 研究目的

基于对学生信息的管理一般有两种方法一种是传统人工方法，另一种是计算机技术存储方法。学生信息管理是学校对学生进行有效管理的一个重要组成部分。它将便于学校快速的对某个学生总体的了解。

管理是一项琐碎，复杂而以十分细致的工作，在过去人们还处于手工管理阶段的时候，那时的管理员摆在他面前和是一大堆的工作无论是信息的录入还是信息的查找都要让他忙上几天几夜，在那个阶段，不要花费管理员的大量时间，而且往往由于抄写不慎，出现张冠李戴的现象。在计算机快速发展的今天，学校把学生的信息都放入了计算机中，为了把学生的在量的信息进行统一的，有效的管理随即产生学生信息管理系统。

1.3 研究意义

在各大中小学校，用计算机管理学校的信息已经越来越普遍了。用计算机不但可以提高工作效率，而且还节省了许多人力物力，增强了学校资料的安全性。提高了学校的管理能力，为此，用计算机来管理学校的信息，是非常必要的。

1.4 研究内容

为了方便各大高校对基层学生的信息管理方便设计了基于Spring MVC和MyBatis框架的Java web项目和部署于阿里云服务器上运行的研究实现方案。本文的主要工作集中在以下几个方面：

* 使用Spring MVC搭配MyBatis框架快速的搭建后台服务，从而实现系统的开发，同时使用MySql数据库来实现平台对于数据方面的操作交互要求。
* 设计并实现学生管理系统的基本功能。用户登录、用户注册、信息的增加、查询、更新、刷新等功能

1.5 本文结构

本论文一共有6章，每章内容介绍如下：

第1章 绪论。介绍学生信息管理系统研究的背景、目的、意义以及内容。

第2章 相关技术综述。介绍Java、Spring MVC和MyBatis框架等相关技术，以及项目开发的环境。

第3章 需求分析。主要介绍系统的需求分析。

第4章 系统架构设计。介绍本系统的架构。

第5章 系统实现。介绍本系统的实现。

第6章 结束语。对自己工作的总结，指出工作的不足

2 相关技术综述

2.1 java

2.1.1 java简介

Java是由Sun Microsystems公司于1995年5月推出的Java面向对象程序设计语言和Java平台的总称。由James Gosling和同事们共同研发，并在1995年正式推出[[[1]](#endnote-1)]。

Java语言发展到今天经历了三个发展方向：J2SE，J2ME，J2EE，最核心的部分是J2SE，J2ME和J2EE是在J2SE基础之上发展起来的，3种技术的关系图如图2-1所示：

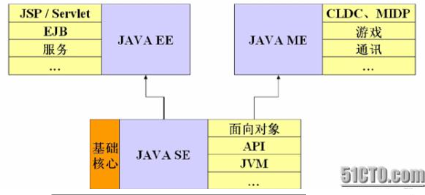


图2-1 J2SE，J2ME和J2EE三者间关系图

2.1.2 java特性

Java语言具有以下特点：

* 简洁有效：Java语言是一种相当简洁的“面向对象”程序设计语言
* 可移植性：java编写的程序可以在多个不同的平台上运行，基本上不需要做任何修改。
* 面向对象：面向对象是指以对象为基本粒度，其下包含属性和方法。Java是一种面向对象的语言。
* 较好的性能：JAVA语言源程序编写完成后，先使用java伪编译器进行伪编译，将其转换为中间码再解释，拥有较好的运行性能。

2.2 Spring MVC框架

2.2.1 Spring MVC框架简介

Spring MVC属于Spring Framework的后续产品，已经融合在Spring Web Flow里面，它原生支持的Spring特性。Spring MVC 分离了控制器、模型对象、分派器以及处理程序对象的角色，这种分离让它们更容易进行定制。SpringMVC在项目中拦截用户请求，它的核心Servlet即DispatcherServlet承担中介或是前台这样的职责，将用户请求通过HandlerMapping去匹配Controller，Controller就是具体对应请求所执行的操作。SpringMVC相当于SSH框架中的struts[[[2]](#endnote-2)]。

2.2.2 Spring MVC框架特性

Spring MVC是一种基于Java，实现了Web MVC设计模式，请求驱动类型的轻量级Web框架，即使用了MVC架构模式的思想，对web层进行解耦，使我们开发更简洁；是Spring的一部分，与Spring无缝衔接；灵活的数据验证，格式化，数据绑定机制。其有以下的一些特点：Spring MVC拥有强大的灵活性、非侵入性和可配置性。Spring MVC提供前端控制器Dispatcher Servlet，开发者无须额外开发。Spring MVC分工明确，包含控制器、处理程序映射视图解析器等，各司其职。Spring MVC支持多视图技术，拥有简单强大的JSP标签库，支持数据绑定[[[3]](#endnote-3)]。

2.3 MyBatis框架

2.3.1 MyBatis框架简介

MyBatis是一个基于Java的持久层框架。mybatis是对jdbc的封装，它让数据库底层操作变的透明。它支持定制化 SQL、存储过程以及高级映射，MyBatis 避免了几乎所有的 JDBC 代码和手动设置参数以及获取结果集[[[4]](#endnote-4)]。几乎避免了JDBC和Hibernate的所有弊端。

2.3.2 MyBatis框架特性

Mybatis有着以下的一些特点：

* 简单易学：没有任何第三方依赖，只要两个jar文件+配置几个sql映射文件，易于使用，通过文档和源代码，可以比较完全的掌握它的设计思路和实现。
* 灵活：mybatis不会对应用程序或者数据库的现有设计强加任何影响。 sql写在xml里，便于统一管理和优化。通过sql语句可以满足操作数据库的所有需求。
* 解除sql与程序代码的耦合：提高了可维护性。
* 提供映射标签，支持对象与数据库的orm字段关系映射
* 提供对象关系映射标签，支持对象关系组建维护
* 提供xml标签，支持编写动态sql。

2.4 开发环境介绍

2.4.1 Eclipse简介

Eclipse允许在同一IDE中集成来自不同供应商的工具，并实现了工具之间的互操作性，从而显著改变了项目工作流程，使开发者可以专注在实际的嵌入式目标上。 Eclipse框架的这种灵活性来源于其扩展点。它们是在XML中定义的已知接口，并充当插件的耦合点。扩展点的范围包括从用在常规表述过滤器中的简单字符串，到一个Java类的描述。任何Eclipse插件定义的扩展点都能够被其它插件使用，反之，任何Eclipse插件也可以遵从其它插件定义的扩展点。除了解由扩展点定义的接口外，插件不知道它们通过扩展点提供的服务将如何被使用。

2.4.2 MySql简介

MySQL是开源多线程的关系数据库管理系统，由Michael “Monty” Winenius在1995年创建的，目前MySQL AB在全球大约有超过6000000的用户[[[5]](#endnote-5)]。

MySQL的插件式存储引擎，将数据库理论发挥到淋漓尽致，完美的映射了数据库的外模式和内模式理论，据MySQL官方称，MySQL数据库占有全球数据库25%的市场份额、互联网公司80%的市场份额[[[6]](#endnote-6)]。

MySQL5.5融合了MySQL数据库和InnoDB存储引擎的优点，能够提供高性能的数据管理解决方案，包括InnoDB作为默认的数据库存储引擎、提高了Windows系统下的系统性能和可扩展性、改善性能和可扩展性、提高实用性。MySQL的优势还在于他的免费开源，能够占据很大的市场比例。

2.4.3 Tomcat8.0简介

Tomcat 服务器是一个免费的开放源代码的Web 应用服务器，属于轻量级应用服务器，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[[[7]](#endnote-7)]。在开发过程中，可以将Java项目部署到Tomcat上，实现模拟效果，极大程度的方便了开发者。在项目开发完成后，也可以将Tomcat作为服务器部署线上的项目。

2.4.4 MVC 设计模式

MVC 是软件工程中的一种架构模式，它的全称是 Model View Controller， 其实即是 Model（模型），View（视图），Controller（控制器）的组合。它将 业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时， 不需要重新编写业务逻辑。 在 Web 应用程序中，MVC 设计模式的三个部分有着各自的业务逻辑： Model（模型层）：Model（模型）是应用程序中用于处理应用程序数据逻辑 的单位对象，通常模型对象负责在数据库中存取数据。举例来说，数据库中每一 个数据表都可以对应一个模型，该模型与数据表拥有相同的属性。在我们需要在 该数据表中获取数据时，该模型就会被使用。 View（视图层）：View（视图）是应用程序处理数据后进行显示的部分，它 提供一个可视化的部分，来较为直观的在用户数据处理完成后产生响应。 Controller（控制层）：Controller(控制器)是应用程序中处理用户交互的部分。 通常控制器负责从视图读取数据，控制用户输入，并向模型发送数据。 MVC 分层有助于管理复杂的应用程序，因为您可以在一个时间内专门关注 一个方面。例如，您可以在不依赖业务逻辑的情况下专注于视图设计。同时也让 应用程序的测试更加容易。 MVC 分层同时也简化了分组开发。不同的开发人员可同时开发视图、控制 器逻辑和业务逻辑。

2.4.5 Spring 框架

Spring 框架用于简化软件开发而创建，是 JavaWeb 开发的常用框架。它的 主要功能是使用 JavaBean 代替 EJB（Enterprise Java Beans 的技术简称，是之前 的应用开发组件）。 在 Spring 框架之前，开发过程中编写的代码经常出现类似或者重复的情 况，这导致开发过程复杂，开发效率和性能都不太理想。Spring 的出现就是为 了解决早期这种应用开发的情况，它致力于提供一个以统一的、高效的方式构 造整个应用，并且可以将单层框架以最佳的组合揉和在一起建立一个连贯的体 系。 Spring 作为一个容器框架，其最大的两个特性是控制反转（IOC）和面向 切面（AOP）。 控制反转：控制反转是 Spring 框架作为容器框架的主要特性。原本我们在 Java 开发中，我们要使用对象时都需要自身创建对象再进行操作。而使用 Spring 框架后，所需对象依赖的其它对象会通过被动的方式传递进来，而不是 这个对象自己创建或者查找依赖对象，方便开发者使用对象的同时节省空间， 促进了松耦合。 面向切面：面向切面是函数式编程的一种衍生范型。利用面向切面可以对 业务逻辑的各个部分进行隔离，从而使得业务逻辑各部分之间的耦合度降低， 提高程序的可重用性，同时提高了开发的效率。 在项目中使用 Spring 框架需要在项目中配置部分的配置文件，一般在项目 中会配置文件搭配注解使用，简化配置并方便进行管理。

3 需求分析

3.1 功能需求分析

学生管理系统面向的用户是学校信息管理人员，人们可以利用此系统来很方便的管理基层学生的信息。该系统可以分为以下几大功能模块：登录注册模块，更新学生信息模块，修改学生信息模块、删除学生信息模块、查询学生信息模块。其功能结构图如图3-1所示。

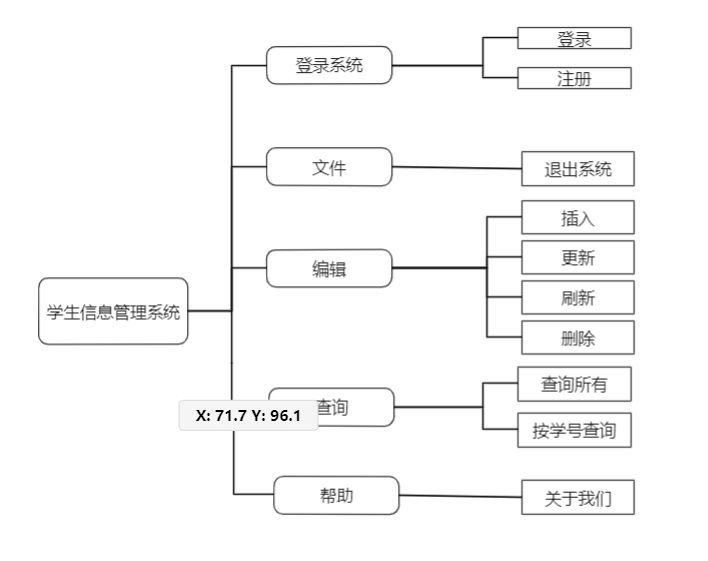


图3-1 学生管理系统功能结构图

4 系统设计

4.1 软件架构设计

学生管理系统整体采用了MVC的架构设计，分为逻辑模型层(Model)、视图层(View)、控制层(Controller）。页面采用Java Swing技术，后端采用了Spring MVC搭配Mybatis持久层框架。逻辑模型层包含对数据库的操作，实体类的定义等，用于向控制层（Controller）提供业务功能接口，向控制层传递数据，提取数据并呈现出来。

4.1 数据库实现

描述各数据表和视图的作用

|  |  |
| --- | --- |
| 数据库名称： | Studentdb |
| 表名 | 描述 |
| student | 学生信息表 |
| user | 管理员表 |

描述重要数据表或视图的结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表名 | student | | | |
| 字段名称 | 描述 | 数据类型 | 字段长度 | 备注 |
| sid | 学号 | int | 8 |  |
| sname | 姓名 | varchar | 20 |  |
| sage | 年龄 | Int | 8 |  |
| shao | 学号 | varchar | 20 |  |
| address | 地址 | varchar | 50 |  |
| sphone | 联系电话 | varchar | 20 |  |
| 表名 | user | | | |
| 字段名称 | 描述 | 数据类型 | 字段长度 | 备注 |
| name | 用户名 | varchar | 20 |  |
| password | 密码 | varchar | 20 |  |

4.2 登录注册功能实现

注册登录等功能都通过Java Swing实现界面的设计，java语言实现后台功能。

4.2.1 登录注册功能实现思路

1、预先设计数据库对应模型，完成数据库的创建

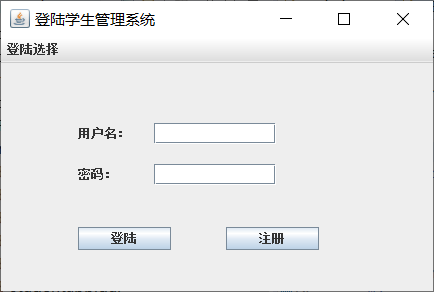
2、连接数据库

3、编写MyBatis的配置文件及其调用的接口,编写之后要使用的sql语句。

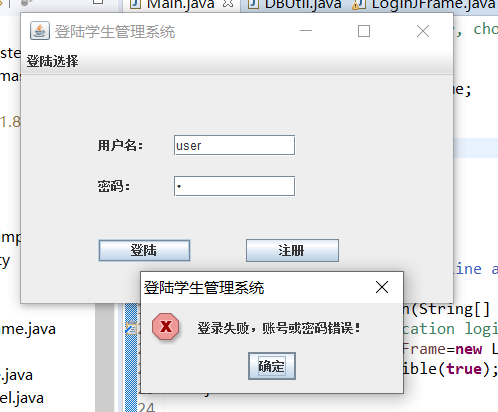
4、使用JavaSwing设计登录界面

5、完成Controller层，处理前端传过来的请求，获取页面传过来的数据并根据需求调用实现写好的sql语句。

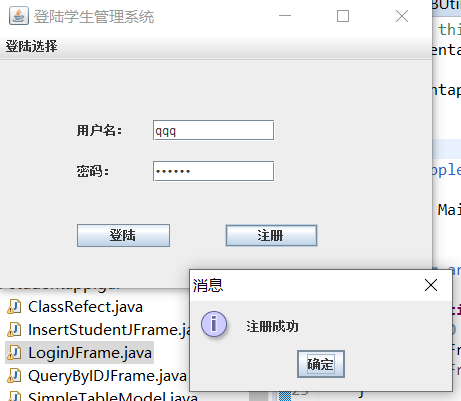
4.2.2 登录注册页面功能实现



登录错误演示：



注册演示：



4.3 学生管理系统主页面功能实现

Java Swing实现界面的设计，java语言实现后台功能。

4.3.1 学生管理系统主页面功能实现思路

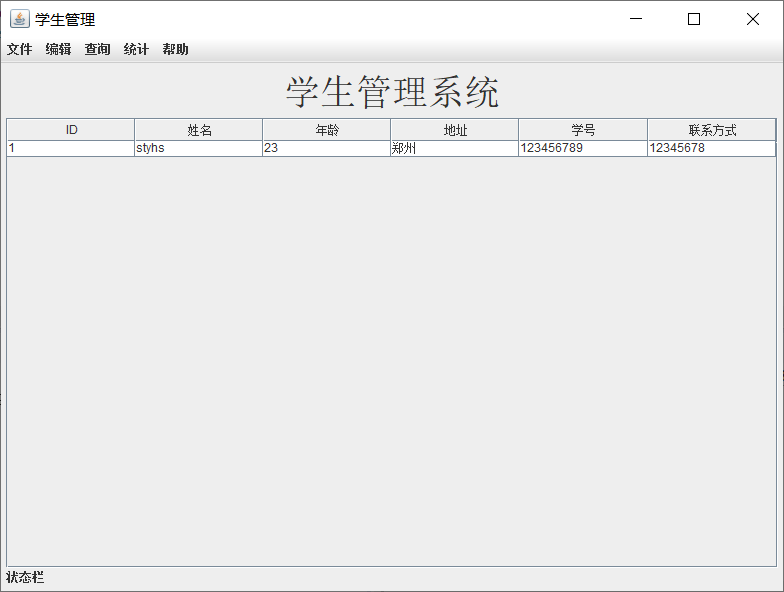
1、连接数据库

2、编写MyBatis的配置文件及其调用的接口,编写之后要使用的sql语句。

3、使用JavaSwing设计主界面

4、完成Controller层，处理前端传过来的请求，获取页面传过来的数据并根据需求调用实现写好的sql语句。

4.3.2 主页面功能实现



4.4 学生信息插入功能实现

4.4.1 学生管理系统学生信息插入功能实现思路

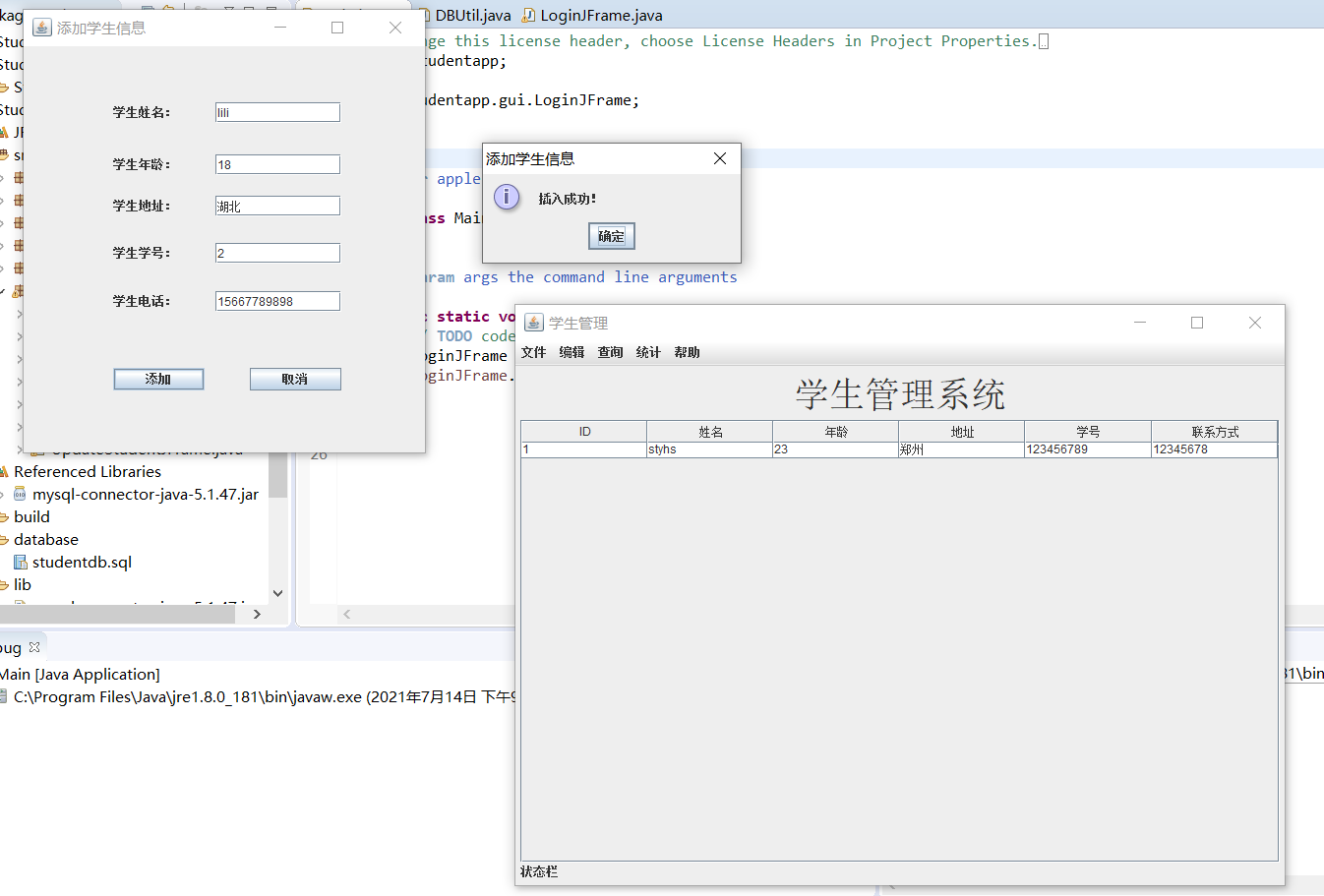
1、实现页面跳转

2、编写MyBatis的配置文件及其调用的接口,编写之后要使用的sql语句。

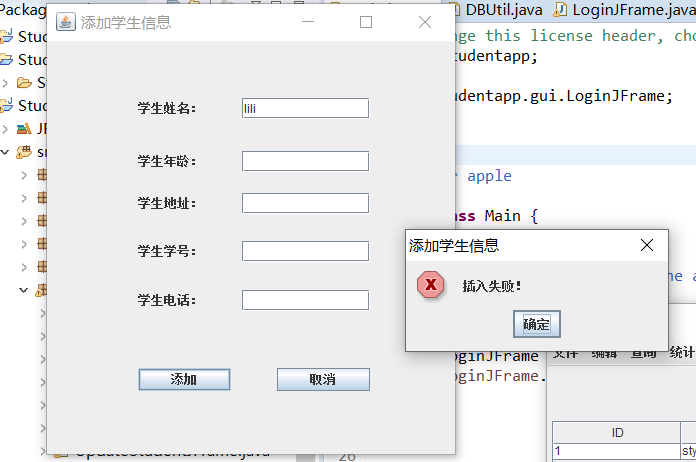
3、使用JavaSwing设计学生信息插入界面

4、完成Controller层，处理前端传过来的请求，获取页面传过来的数据并根据需求调用实现写好的sql语句。

4.4.2 学生信息插入页面功能实现



插入失败演示：



4.5 学生信息更新功能实现

4.5.1 学生管理系统学生信息更新功能实现思路

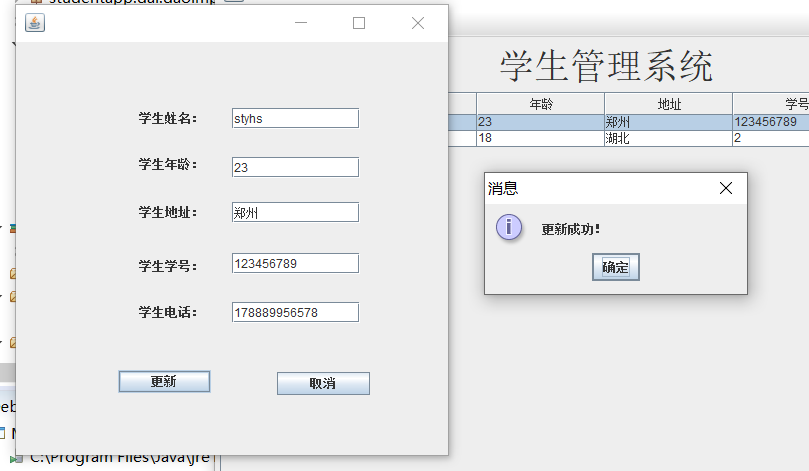
1、实现页面跳转

2、编写MyBatis的配置文件及其调用的接口,编写之后要使用的sql语句。

3、使用JavaSwing设计学生信息更新界面

4、完成Controller层，处理前端传过来的请求，获取页面传过来的数据并根据需求调用实现写好的sql语句。

4.5.2 学生信息更新页面功能实现



4.6 学生信息刷新功能实现

4.6.1 学生管理系统学生信息刷新功能实现思路

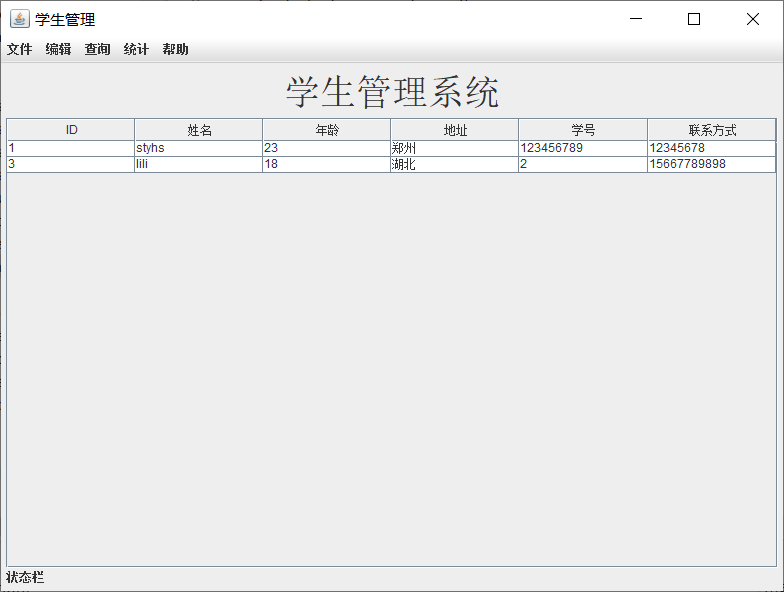
1、实现页面跳转

2、编写MyBatis的配置文件及其调用的接口,编写之后要使用的sql语句。

3、使用JavaSwing设计学生信息刷新界面

4、完成Controller层，处理前端传过来的请求，获取页面传过来的数据并根据需求调用实现写好的sql语句。

4.6.2 学生信息更新页面功能实现





4.7 学生信息删除功能实现

4.7.1 学生管理系统学生信息删除功能实现思路

1、实现页面跳转

2、编写MyBatis的配置文件及其调用的接口,编写之后要使用的sql语句。

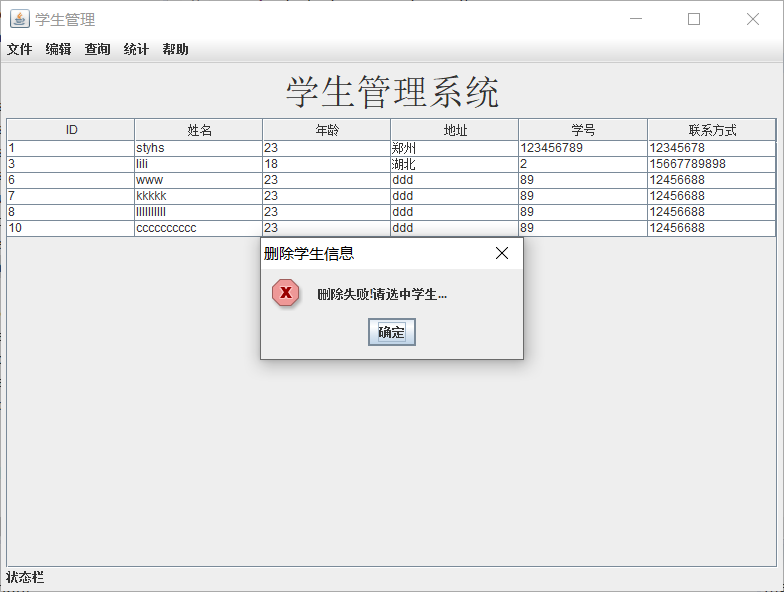
3、完成Controller层，处理前端传过来的请求，获取页面传过来的数据并根据需求调用实现写好的sql语句。

4、调用showMessageDialog事件弹出相应的弹窗显示信息

4.7.2 学生信息更新页面功能实现



删除失败演示：



4.8 学生管理系统查询功能实现

Java Swing实现界面的设计，java语言实现后台功能。

4.8.1.1 学生管理系统查询所有功能实现思路

1、连接数据库

2、完成Controller层，处理前端传过来的请求，获取页面传过来的数据并根据需求调用实现写好的sql语句。

4.8.1.2 查询所有功能实现

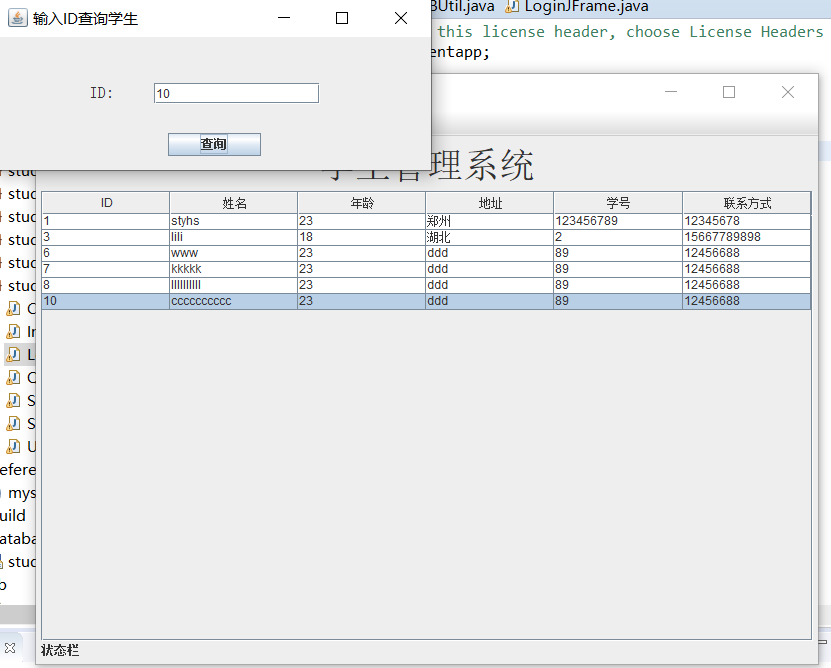


4.8.2.1 学生管理系统ID查询功能实现思路

1、连接数据库

2、完成Controller层，处理前端传过来的请求，获取页面传过来的数据并根据需求调用实现写好的sql语句。

4.8.2.2 ID查询功能实现



4.9 帮助功能实现

4.7.1 学生管理系统帮助功能实现思路

1、调用showMessageDialog事件弹出相应的弹窗显示信息



5 系统实现

5.1 开发环境搭建

“网页云收藏管理平台”先使用java语言在本地开发，然后再部署到阿里云服务器上运行。所以其开发环境包括在本地开发所需环境的实现和阿里云服务器环境的搭建两大部分。

5.1.1 java环境

一、Java环境实现需要以下软件：windows 10操作系统；JDK

1. 安装步骤如下：

第一步：下载jdk；下载网站http://www.oracle.com/ ；

第二步：配置jdk环境变量；在“系统变量”中设置3项属性PATH，JAVA\_HOME,CLASSPATH。

变量设置参数如下：

变量名：JAVA\_HOME

变量值：D:\Java\jdk1.7.0\_17 //根据自己实际路径配置

变量名：CLASSPATH

变量值：.;%JAVA\_HOME%\lib\dt.jar;%JAVA\_HOME%\lib\tools.jar;//记得前面有个"."

变量名：PATH

变量值：%JAVA\_HOME%\bin;%JAVA\_HOME%\jre\bin;

5.1.2 MySQL数据库环境搭建

下载地址：https://www.mysql.com/downloads/。

安装：双击应用程序，一般默认完成安装，成功安装测试界面如图5-3所示。

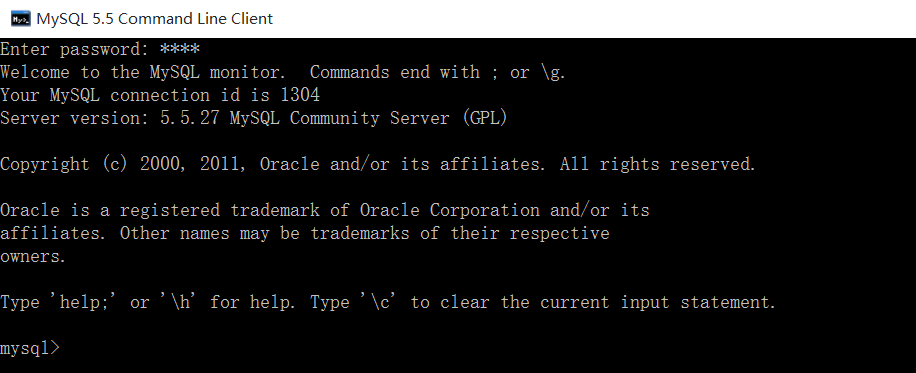


图5-3 MySql安装成功测试界面

6 结束语

由于对java知识的掌握不是很熟练，所以写代码时很艰难，一开始的时候不知道怎么布局。起初在设计的时候，只知道组件的功能以及作用，但是添加上去的部件怎么去给它固定，排版还有对于点击事件的处理不是很清楚的。对于第一个问题，这个问题是在看完书后解决的，上面清楚的说明了如何布局。但是一开始还是有点不是很清楚，于是详细的细细研究，发现可以先设置一个面板为内容面板，然后在内容面板上面添加其他的小面板，再在小面板上面添加所需要的控件（可以这么理解，先在frame上面放一个大布，这个布覆盖了整个窗口，再在这个大布上面放其他的小布，至于小布怎么摆放就是要看大布的布局方式了，然后再在小布上面添加所需要的部件，为了可以让部件看着特别完整美观，可以进一步对于小布进行布局的设置，这样层层嵌套，就可以很完整的把布局整好了，而且外观整齐大方）；对于第二个问题，这个问题是百度了一下，虽说当时百度上面说的特别简单，只有一句话，就是关闭当前窗口，打开新窗口。不过这个问题很简单的就解决了。接着就是对数据了的链接，怎么也链接不上去，一开始是直接本机IP地址链接，后来觉得最后汇总会非常麻烦，然后就改成了直接识别本机的数据库，对工程的可用性的提高很有作用。对于注册功能的实现没有设置好，对于已经存在的用户还能注册，没有设置好sql语句。对于删除功能的实现没有让用户再次确认，影响用户使用感。导入导出功能没有实现，程序功能还能再次完善。

这次的过程的小艰辛只有我才知道，一开始只是知道一些比较零碎的知识，而且很理论化，没有把这些理论化为实践，有句话说得好，实践是检验真理的唯一标准。出现的问题自己在书中和网络上面找到了解决办法，就带马上解决，一刻都不能耽误。让我加深了对Javaweb理论知识的理解，其次，让我更加熟练的掌握了Javaweb语言的设计能力并且有了实践经验，最后，加强了我对Javaweb语的学习兴趣和好感。

参考文献

1. [] 李兴华.Java开发实战经典[M].清华大学出版社.2009. [↑](#endnote-ref-1)
2. [] 任师艳.Spring MVC[OL].https://www.jianshu.com/p/3762c87e4830,2017-11-20. [↑](#endnote-ref-2)
3. [] 疯狂软件.Spring+MyBatis企业应用实战[M].电子工业出版社.2017. [↑](#endnote-ref-3)
4. [] 杨开振.深入浅出MyBatis技术原理与实战[M].电子工业出版社.2016. [↑](#endnote-ref-4)
5. [] Russell J.T.Dyer .MySQL in a Nutshell, 2nd Edition[M].机械工业出版社.2008. [↑](#endnote-ref-5)
6. [] 朱定泽.MySQL核心内幕[M].清华大学出版社.2010. [↑](#endnote-ref-6)
7. [] cms.Java advanced architecture interview Zhenti summary: Nginx, Netty, Zookeeper, Tomcat, Dubbo[OL].https://bzdww.com/article/288642/,2019-04-14. [↑](#endnote-ref-7)