武汉纺织大学

Java应用开发课程设计

**基于MyBatis的超市收银系统**

**学 院： 数计学院**

**班 级： 软件11801**

**姓 名： 梁亚玲**

**学 号： 1814220201**

**指导老师： 聂刚老师**

**成 绩：**

**完成日期： 2020年6月20日**

# 1 需求分析

采用MyBatis框架实现基于DAO设计模式的学生选课管理系统：

角色划分为：学生、教师、管理员

具体功能实现为：

**1.1用户登录**

当程序运行是，显示“欢迎使用武汉纺织大学教务管理系统，请登录！”，提示用户输入用户名及密码以完成登录，当用户名及密码输入错误时，退出系统；

**1.2主菜单**

当登录完成后，用户进入主菜单，界面如下：

-----------------------------------------------------

武汉纺织大学教务管理系统

1.我是学生——>进入学生管理系统

2.我是教师——>进入教师管理系统

3.我是管理员——>进入用户管理系统

4.修改账户密码

5.退出

请输入:

当输入不合法时，弹出“输入无效！请重新输入:”，提示用户重新进行选择。

**1.3学生管理系统**

当前用户角色不为“学生”时，弹出提示“无权限进入！您当前的身份不是学生！”。用户选择并进行判断角色身份为“学生”后，进入学生管理系统，并弹出学生管理系统菜单，如下：

-----------------------------------------------------

武汉纺织大学教务管理系统——>您当前所处位置：学生管理系统

1.查看当前已选课程

2.选课

3.退课

4.查询个人成绩

5.返回主菜单

请输入:

**1.3.1 查看当前已选课程**

角色身份为“学生”的用户进入后，在选择1时，可查看当前已选课程，输出格式为：

姓名 学号 课程编号 课程名 任课教师 学时

==== ==== ======= ===== ====== ====

**1.3.2选课**

角色身份为“学生”的用户进入后，在选择2时，可进行选课，从键盘依次录入选课信息，格式为“姓名，教师id，教师姓名，课程编号，课程名，课程学时”，选课成功后弹出提示“选课成功！”。

**1.3.3退课**

角色身份为“学生”的用户进入后，在选择3时，可进行退课，根据系统提示“请输入退选课程的课程编号:”，从键盘录入信息后，退课成功弹出消息提示“退课成功”。

**1.3.4查询个人成绩**

角色身份为“学生”的用户进入后，在选择4时，可查询个人成绩，输出格式为：

姓名 学号 课程编号 课程名 成绩

==== ==== ======= ===== ===

**1.3.5返回主菜单**

角色身份为“学生”的用户进入后，在选择5时，即可返回主菜单并重新进行选择。

**1.4教师管理系统**

当前用户角色不为“教师”时，弹出提示“无权限进入！您当前的身份不是教师！”。用户选择并进行判断角色身份为“教师”后，进入教师管理系统，并弹出教师管理系统菜单，如下：

-----------------------------------------------------

武汉纺织大学教务管理系统——>您当前所处位置：教师管理系统

1.查看我的课程

2.查看选课名单

3.导出选课名单

4.录入学生成绩

5.导出学生成绩

6.返回主菜单

请输入:

**1.4.1查看我的课程**

角色身份为“教师”的用户进入后，在选择1时，可查看当前个人的授课信息，输出格式为：

教师姓名 课程编号 课程名 学时

======= ====== ===== ====

**1.4.2查看选课名单**

角色身份为“教师”的用户进入后，在选择2时，根据提示“请输入课程编号:”输入具体的课程编号后可查看当前课程的选课名单，输出格式为：

学生姓名 学号

======= ===

**1.4.3导出选课名单**

角色身份为“教师”的用户进入后，在选择3时，在输入课程编号后，将当前课程的选课明道导出为xls文件，导出的文件名为“SelectionDetail.xls”，并在成功导出后提示“成功导出\*条数据到excel文件中”。

**1.4.4录入学生成绩**

角色身份为“教师”的用户进入后，在选择4时，即可录入学生成绩，从键盘依次输入成绩信息，格式为“课程编号，课程名称，学生学号，学生姓名，课程成绩”，录入完成后提示“成绩录入成功！”。

**1.4.5导出学生成绩**

角色身份为“教师”的用户进入后，在选择5时，在输入课程编号后，将当前课程的选课明道导出为xls文件，导出的文件名为“StudentMark.xls”，并在成功导出后提示“成功导出\*条数据到excel文件中”。

**1.4.6返回主菜单**

角色身份为“教师”的用户进入后，在选择6时，即可返回主菜单并重新进行选择。

**1.5用户管理系统**

当前用户角色不为“管理员”时，弹出提示“无权限进入！您当前的身份不是管理员！”。用户选择并进行判断角色身份为“管理员”后，进入学生管理系统，并弹出用户管理系统菜单，如下：

-----------------------------------------------------

武汉纺织大学教务管理系统——>您当前所处位置：用户管理系统

1.增加用户信息

2.删除用户信息

3.增加课程信息

4.删除课程信息

5.返回主菜单

请输入:

**1.5.1增加用户信息**

角色身份为“管理员”的用户进入后，在选择1时，即可添加用户信息，从键盘依次录入信息，输入格式为“用户名，id，密码，角色”，添加完成后提示“信息添加成功！”。

**1.5.2删除用户信息**

角色身份为“管理员”的用户进入后，在选择2时，根据提示“请输入删除的账号id:”输入账号id后，即可完成用户的删除，在删除成功后提示“信息删除成功！”。

**1.5.3增加课程信息**

角色身份为“管理员”的用户进入后，在选择3时，即可增加课程信息，从键盘依次录入信息，输入格式为“教师id，教师姓名，课程编号，课程名,课程学时”，添加完成后提示“课程添加成功！”。

**1.5.4删除课程信息**

角色身份为“管理员”的用户进入后，在选择4时，根据提示“请输入删除的课程编号:”输入课程编号后，即可完成课程的删除，在删除成功后提示“课程删除成功！”。

**1.5.5返回主菜单**

角色身份为“管理员”的用户进入后，在选择6时，即可返回主菜单并重新进行选择。

**1.6修改账户密码**

当选择4时，显示“请输入当前用户的原密码：”，若原密码不正确，则提示“密码不正确!”，并返回主菜单，输入正确后，显示“请输入新的密码:”，输入完成后要求再次确认密码，俩次输入一致则密码更改成功，提示“修改成功!”并返回主菜单。

**1.7退出**

选择5时，显示“感谢您的使用！”并退出系统。

# 2 系统设计

**2.1用户用例图**

系统分为学生，教师，管理员三个角色，各角色的用户用例如下图2-1所示：

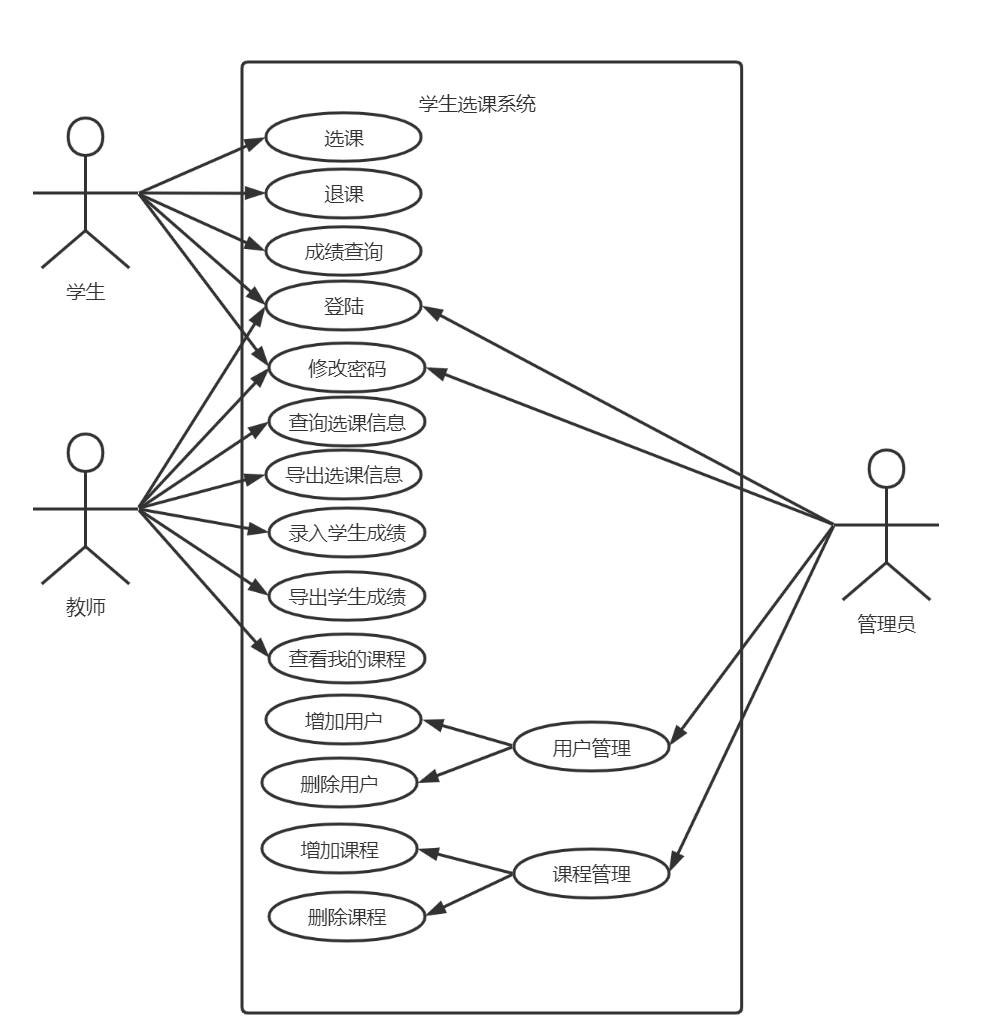


图2-1 用户用例图

## 2.2 ER图

数据库设计4张表，用户表，成绩表，课程表，选课记录表，对应的ER图如图3-1所示。

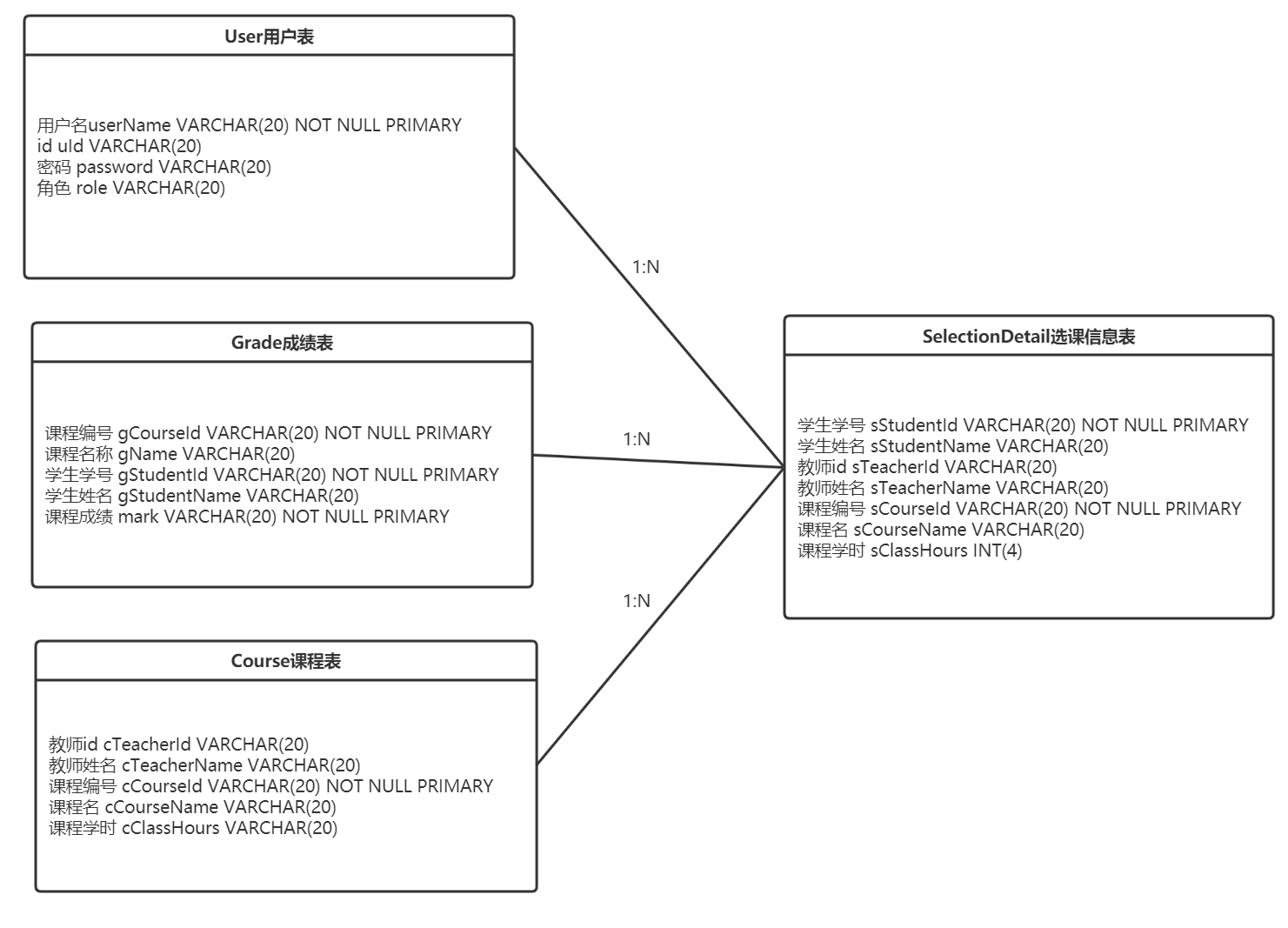


图3-1 数据库ER图

## 2.3 UML类图（Class Diagram）

对于用户功能模块，共设计如下5个类。

* VO实体类：与数据库结构进行映射的类。主要由属性，setter, getter方法组成，VO类中的属性与表中的字段相对应，每一个VO类的对象都表示表中的每一条记录
* DAO接口：主要定义操作的接口，定义一系列数据库的原子性操作，例如增删改查（通常称为CRUD）等。
* 映射mapper：主要完成具体数据库操作。
* Config配置文件： mybatis配置。
* Driver：执行具体操作。
* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*：

各类的结构及类之间的关系如图2-5所示：

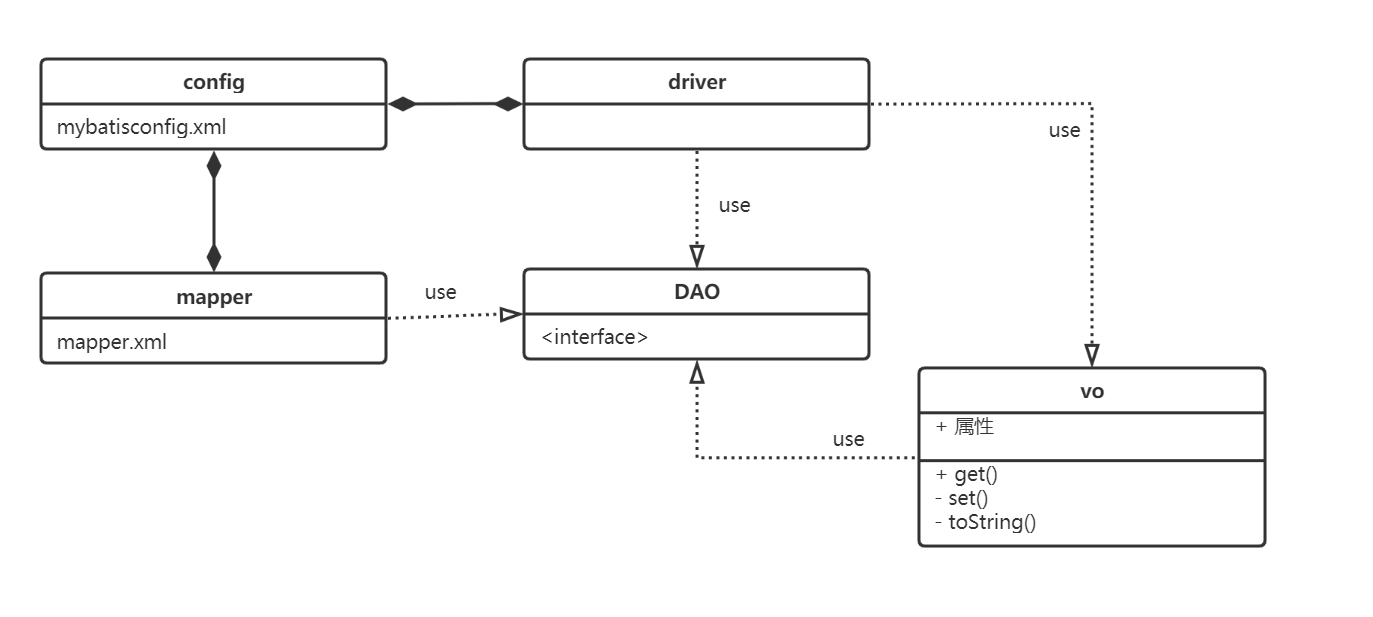
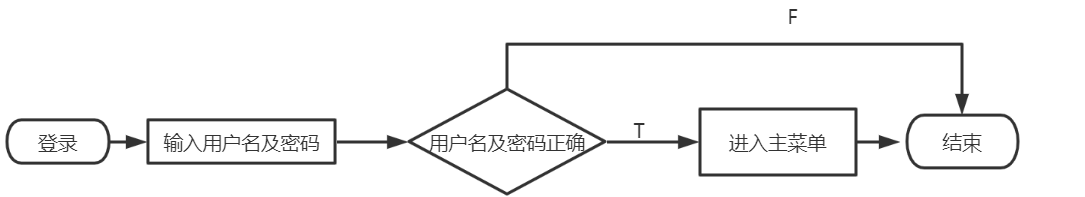


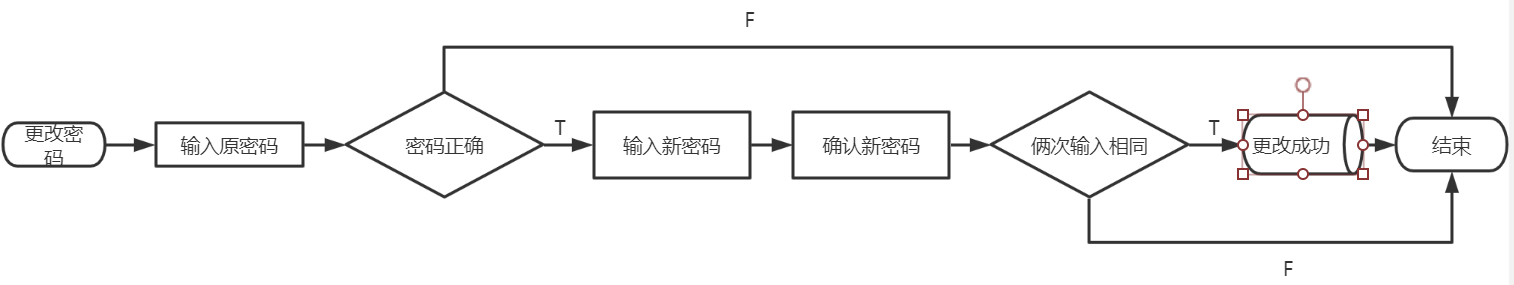
图2-5 用户功能类图

## 2.5流程图

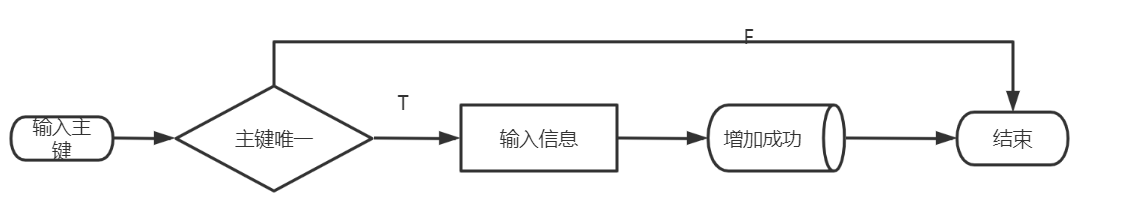
用户登录



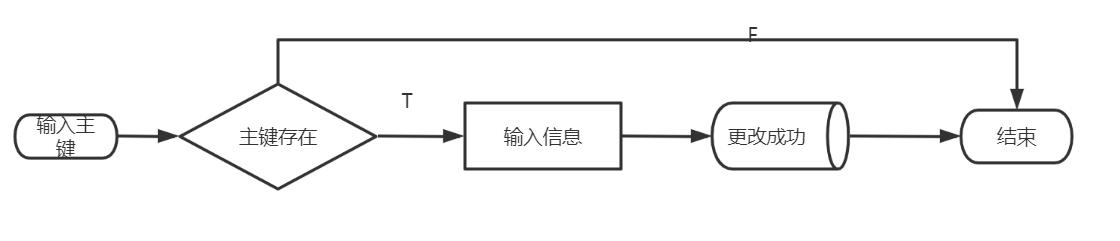
更改密码



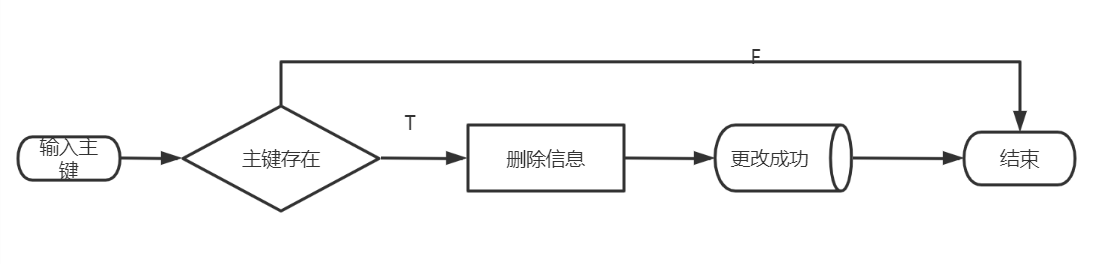
增加信息



更改信息

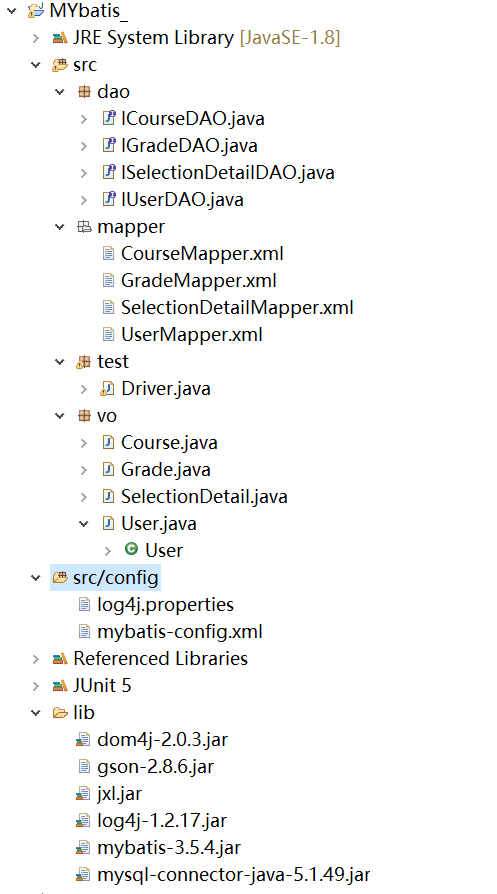


删除信息



# 3 系统实现

## 3.1 项目结构



## 3.2 配置文件

在项目下新建Source Folder，名为config

### 3.2.1 log4j.properties文件

log4j.rootLogger=DEBUG, stdout

log4j.appender.stdout=org.apache.log4j.ConsoleAppender

log4j.appender.stdout.layout=org.apache.log4j.PatternLayout

log4j.appender.stdout.layout.ConversionPattern=%5p [%t] - %m%n

### 3.2.3 mybatis-config.xml文件

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE configuration

PUBLIC "-//mybatis.org//DTD Config 3.0//EN"

"http://mybatis.org/dtd/mybatis-3-config.dtd">

<!-- 根标签 -->

<configuration>

<!-- 设置Mybatis的log实现为LOG4J -->

<settings>

<setting name="logImpl" value="LOG4J" />

</settings>

<!-- 环境，可以配置多个，default：指定采用哪个环境 -->

<environments default="test">

<!-- id：唯一标识 -->

<environment id="test">

<!-- 事务管理器，JDBC类型的事务管理器 -->

<transactionManager type="JDBC" />

<!-- 数据源，池类型的数据源 -->

<dataSource type="POOLED">

<property name="driver" value="com.mysql.jdbc.Driver"/>

<property name="url" value="jdbc:mysql://127.0.0.1:3306/school?useUnicode=true&amp;characterEncoding=UTF-8" />

<property name="username" value="root"/>

<property name="password" value="Lyl99999"/>

</dataSource>

</environment>

</environments>

<!-- 配置mapper映射文件 -->

<mappers>

<mapper resource="mapper/CourseMapper.xml" />

<mapper resource="mapper/UserMapper.xml" />

<mapper resource="mapper/GradeMapper.xml" />

<mapper resource="mapper/SelectionDetailMapper.xml" />

</mappers>

</configuration>

## 3.3 VO类User.java

public class User {

private String userName;//用户名

private String uId;// id

private String password;//密码

private String role;//角色

public User() {

super();

}

public User(String userName, String uId, String password, String role) {

super();

this.userName = userName;

this.uId = uId;

this.password = password;

this.role = role;

}

public String getUserName() {

return userName;

}

public void setUserName(String userName) {

this.userName = userName;

}

public String getuId() {

return uId;

}

public void setuId(String uId) {

this.uId = uId;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

public String getRole() {

return role;

}

public void setRole(String role) {

this.role = role;

}

@Override

public String toString() {

return "User [userName=" + userName + ", uId=" + uId + ", password=" + password + ", role=" + role + "]";

}

## 3.4 DAO接口类IUserDAO.java

public interface IUserDAO {

//增加记录

public boolean insert(User user) throws Exception;

//根据主键(id)删除记录

public boolean delete(String uId) throws Exception;

//根据主键(id)更改信息

public boolean update(User user) throws Exception;

//根据主键(id)查询信息

public User find(String uId) throws Exception;

//根据条件查询信息:查询条件封装在user对象中，若user对象的某个成员变量值为null，则表示查询时忽略该字段查询条件

public List<User> query(User user) throws Exception;

}

## 3.5 接口映射文件UserMapper.xml

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"

"http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="dao.IUserDAO">

<insert id="insert">

insert INTO user(userName,uId,password,role) VALUES (#{userName},#{uId},#{password},#{role})

</insert>

<delete id="delete">

delete from user WHERE uId=#{uId}

</delete>

<update id="update">

update user SET password=#{password} WHERE uId=#{uId}

</update>

<select id="find" resultType="vo.User">

select \* from user where uId=#{uId}

</select>

<select id="query" resultType="vo.User">

select \* from user

<where>

<if test="userName !=null and userName!=''">

userName =#{userName}

</if>

<if test="uId !=null and uId !=''">

AND uId =#{uId}

</if>

<if test="password !=null and password!=''">

AND password=#{password}

</if>

<if test="role !=null and role!=''">

AND role=#{role}

</if>

</where>

</select>

</mapper>

## 3.8 用户界面Driver.java

public interface IUserDAO {

//增加记录

public boolean insert(User user) throws Exception;

//根据主键(id)删除记录

public boolean delete(String uId) throws Exception;

//根据主键(id)更改信息

public boolean update(User user) throws Exception;

//根据主键(id)查询信息

public User find(String uId) throws Exception;

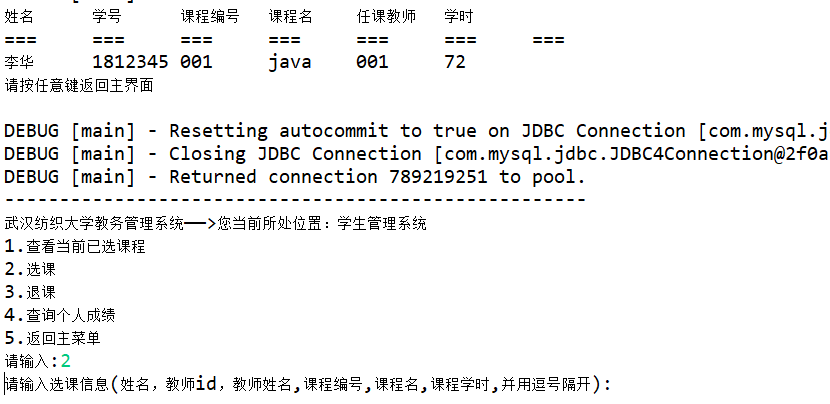
//根据条件查询信息:查询条件封装在user对象中，若user对象的某个成员变量值为null，则表示查询时忽略该字段查询条件

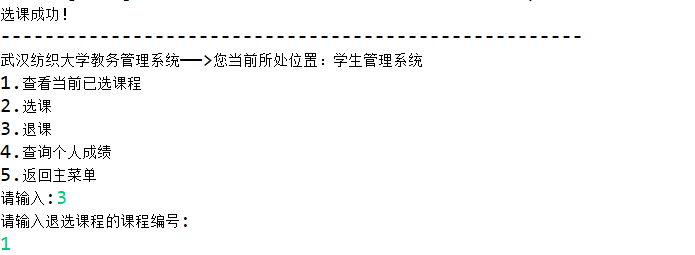
public List<User> query(User user) throws Exception;

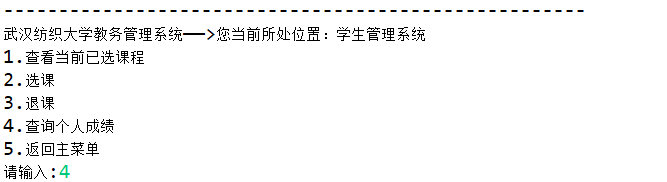
}

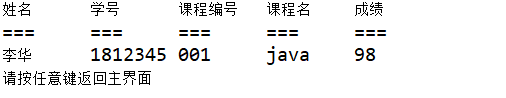
# 4 系统测试

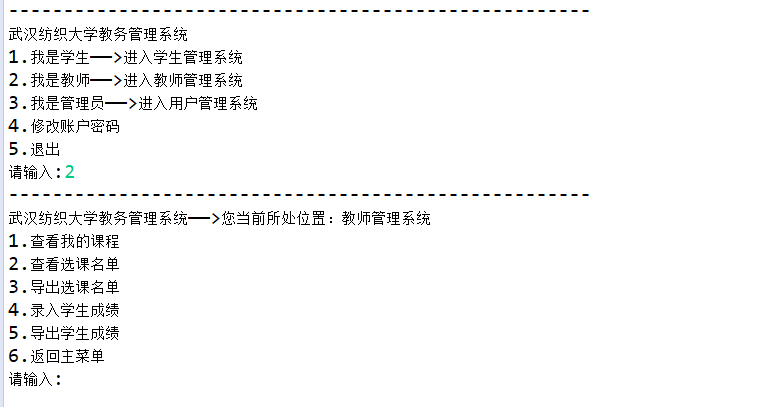


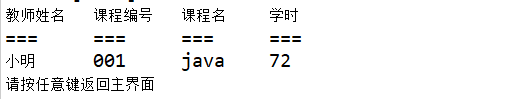


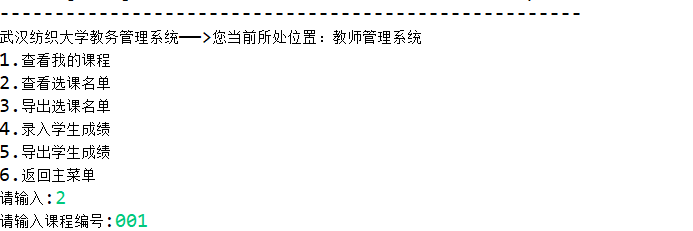


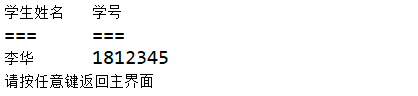


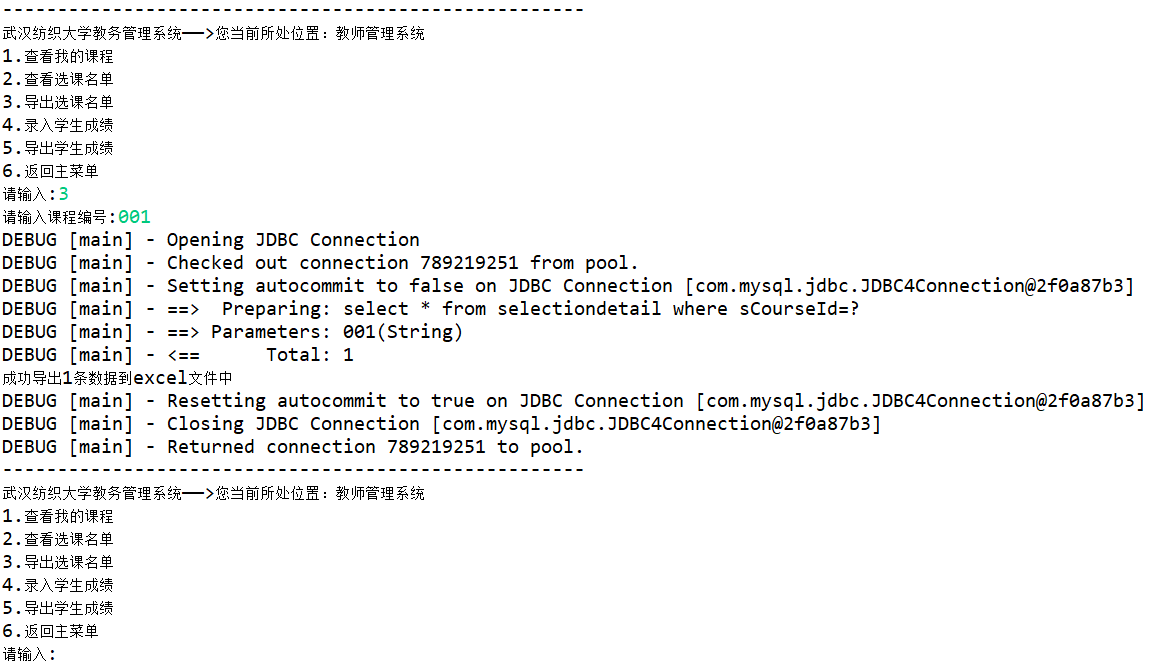




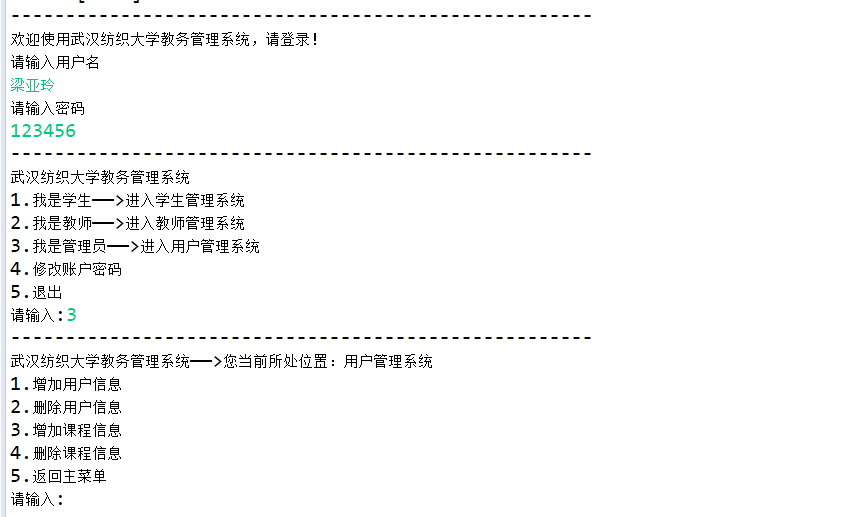


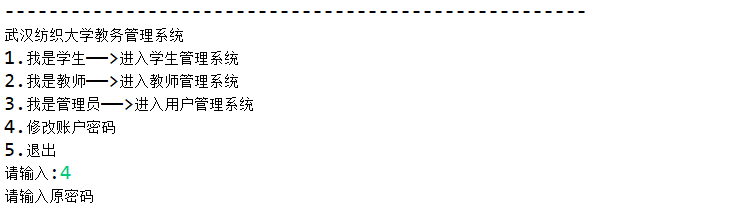


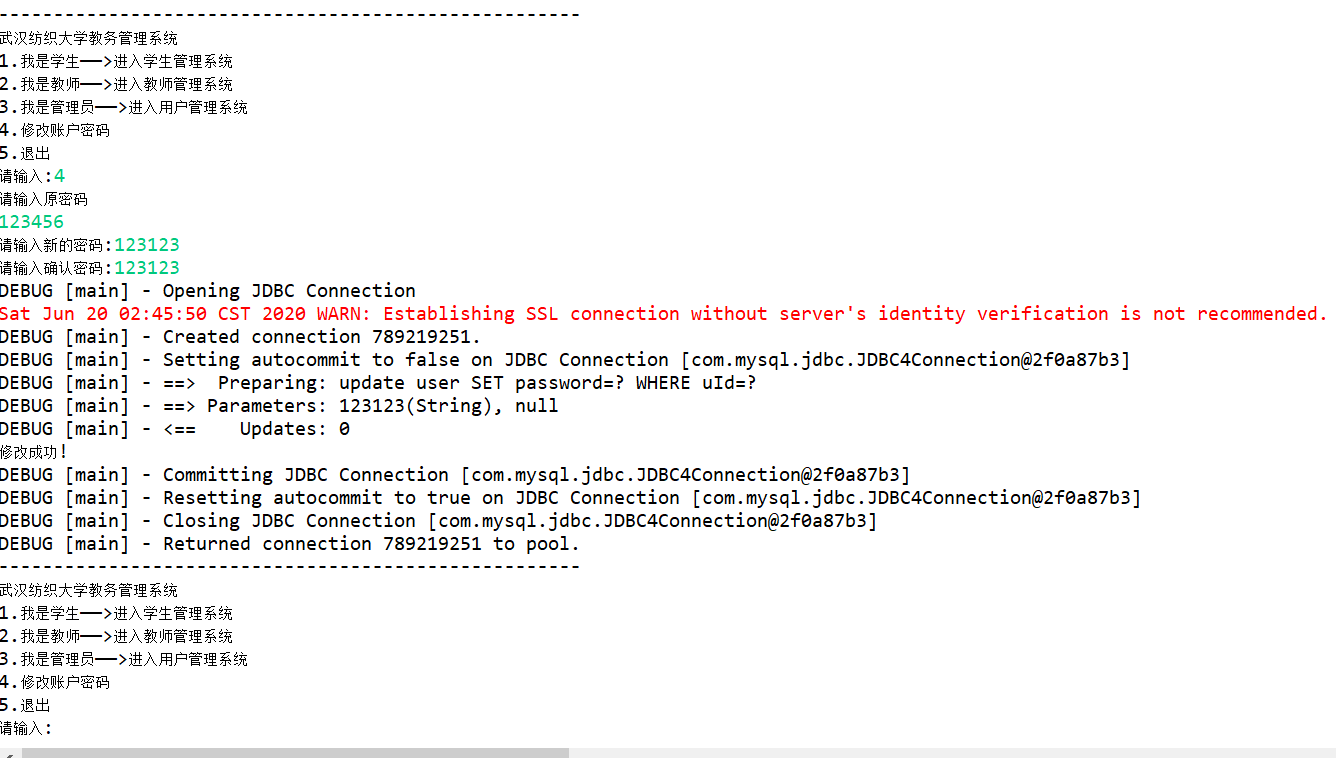


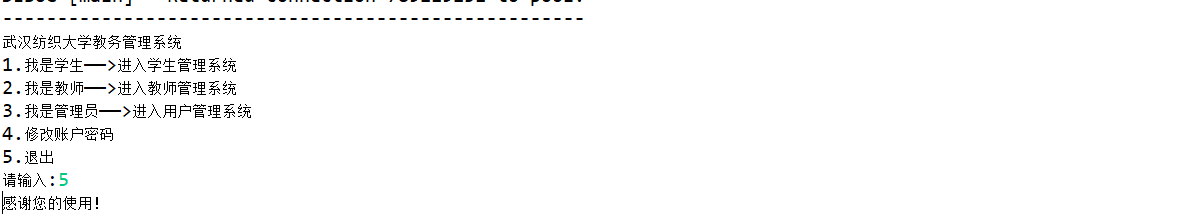












# 5 系统总结

**主要问题及解决过程:**

在完成系统时主要遇到的问题集中于mybatis.xml与mapper映射文件上，“Error parsing SQL Mapper Configuration”“Error parsing SQL Mapper Configuration”等问题，查阅相关资料后发现大多是与mapper映射文件与配置文件的相关错误，如文件名不统一，文件路径等问题，但大多能够通过查阅资料的方式来解决。

**系统存在的不足：**

尽管系统完成基本功能，但在具体的实现上仍存在较大的弊端，如教师端录入学生成绩需要同时录入学生的学号等个人信息以及在录入成绩时只能通过多次重复的选择并录入、学生端选课时无法当前可选课程等。

**课程收获：**

虽然系统仍然存在遗憾，但我仍然觉得能够完成基本的功能实现对已经是很大的进步。很难想象三个月前我们才刚刚开始接触Java语言，在三个月的学习中，虽然由于疫情的影响只能通过线上的学习，但是仍然学习到了、收获到了很多知识：从Java的基本语法到面向对象再到Java的类集，数据库开发最后深入到框架的学习。

**自我评价：**

对我个人来说，线上学习是一个比较痛苦的过程，在线上直播课时往往在不留神之间就开起了小差，但是由于前期Java的基本语法及面向对象等内容难度并没有特别大，还能够通过后期的教案学习等方式完成作业。当进入到数据库的连接时，知识的深入使我不得不在课下使用更长的时间去学习，在课后不断地看回放、看教案才能够理解和掌握，作业也从能够提前完成到勉强赶在截止日期前完成。其实也值得庆幸的是，在这一特殊阶段的学习里我并没有放弃，而是坚持到了最后，所以才得以顺利的完成了期末的大作业。但希望在下学期的Java的第二阶段学习能够保持时刻不掉队呀！！