Xxxxxxxxxxxxxxxxxxx

本科毕业设计

基于SpringBoot框架的猫猫影院购票系统设计与实现

|  |  |
| --- | --- |
| 院部名称 | 信息工程学院 |
| 姓　　名 | Xxx |
| 学　　号 | xxxx |
| 专　　业 | 软件工程 |
| 届　　别 | xxx |
| 指导教师 | xxx |

2024年4月13日

基于SPRINGBOOT框架的影院购票系统设计与实现

**摘要：**伴随着互联网的快速发展，网络技术被普遍的应用在了我们生活的各个领域。传统的电影购票方式，需要用户去电影院才能了解到上映的电影的信息，这对于工作繁忙的上班族来说是很不方便的。影院购票系统可以让用户随时随地快速的了解到上映电影的最新信息，在方便用户购票的同时也节省了电影院的支出。该系统使用Java语言进行开发，采用SpringBooot框架，结合MySQL数据库设计搭建了集合前台购票和后台管理为一体的影院购票系统。系统分为前台购票界面和后台管理界面，前台购票界面，为用户提供了用户注册、商品浏览、个人中心、电影详情介绍等主要功能。影院购票系统的后台管理界面，为管理员提供了很多的管理功能，以便于管理员对管理员的基本信息、电影的基本信息、用户信息、用户所创建的订单信息、电影信息等进行管理。

**关键词：**SpringBooot；影院购票；Java；MySQL

Design and Implementation of Cinema Booking System Based on SPRINGBOOT Framework

**Abstract:** With the rapid development of the Internet, network technology has been widely used in all areas of our lives. The traditional way of purchasing movie tickets requires users to go to the cinema to learn about the information of the released movies, which is very inconvenient for busy office workers. The cinema ticketing system allows users to quickly and anytime stay informed about the latest information about movies being released, making it convenient for users to purchase tickets while also saving on cinema expenses. The system was developed using Java language and the SpringBoot framework, combined with MySQL database design, to build a cinema ticketing system that integrates front-end ticketing and back-end management. The system is divided into a front-end ticketing interface and a back-end management interface. The front-end ticketing interface provides users with main functions such as user registration, product browsing, personal center, and movie details introduction. The backend management interface of the cinema ticketing system provides administrators with many management functions, making it easy for administrators to manage basic information of administrators, movies, user information, order information created by users, movie information, etc.

**Keywords:** SpringBooot;Theater Booking;Java;MySQL

目录

[1 绪论 1](#_Toc7916)

[1.1 研究背景和意义 1](#_Toc26662)

[1.1.1 研究背景 1](#_Toc21620)

[1.1.2 研究意义 1](#_Toc12108)

[1.2 国内外文献综述 1](#_Toc6254)

[1.3 研究的主要内容和方法 2](#_Toc9525)

[1.3.1 研究内容 2](#_Toc18120)

[1.3.2 研究方法 3](#_Toc26888)

[1.4 所用技术介绍 3](#_Toc4430)

[1.4.1 Java编程语言介绍 3](#_Toc16280)

[1.4.2 SPRINGBOOT介绍 3](#_Toc732)

[1.4.3 MySQL数据库介绍 4](#_Toc17499)

[2 系统分析 6](#_Toc15597)

[2.1 可行性分析 6](#_Toc16186)

[2.1.1 经济可行性 6](#_Toc19131)

[2.1.2 技术可行性 6](#_Toc11697)

[2.1.3 社会因素可行性 6](#_Toc32337)

[2.2 需求分析 6](#_Toc16919)

[2.2.1 功能需求分析 6](#_Toc12095)

[2.2.2 性能需求分析 8](#_Toc29195)

[3 系统设计 9](#_Toc12236)

[3.1 系统结构设计 9](#_Toc13649)

[3.2 功能模块设计 9](#_Toc13528)

[3.3 数据库设计 12](#_Toc24751)

[3.3.1 数据库概念设计 12](#_Toc3544)

[3.3.2 数据库逻辑设计 12](#_Toc9344)

[4 系统实现 14](#_Toc13705)

[4.1 系统前台界面的实现 14](#_Toc13597)

[4.1.1 用户注册登录功能的实现 14](#_Toc18758)

[4.1.2 用户订单管理功能的实现 14](#_Toc4938)

[4.1.3 用户信息管理功能的实现 15](#_Toc9267)

[4.1.4 用户账户密码管理功能的实现 15](#_Toc23878)

[4.1.5 用户电影评论功能的实现 16](#_Toc23878)

[4.2 系统后台界面的实现 16](#_Toc1176)

[4.2.1 管理员登录功能的实现 16](#_Toc5429)

[4.2.2 管理员修改密码功能的实现 17](#_Toc5398)

[4.2.3 电影信息管理功能的实现 17](#_Toc28665)

[4.2.4 订单管理功能的实现 18](#_Toc16555)

[4.2.5 电影详情介绍管理功能的实现 18](#_Toc28665)

[4.2.6 影院公告管理功能的实现 19](#_Toc16555)

[5 系统测试 20](#_Toc6072)

[5.1 测试目的及方法 20](#_Toc12181)

[5.2 测试用例 20](#_Toc26526)

[5.2.1 系统前台功能性测试 20](#_Toc31897)

[5.2.2 系统后台功能性测试 21](#_Toc4760)

[5.3 测试结论 21](#_Toc27961)

[结论 23](#_Toc29990)

[参考文献 24](#_Toc3293)

[致谢 25](#_Toc4722)

# 1 绪论

1.1 研究背景和意义

1.1.1 研究背景

随着科技与计算机网络技术的快速发展，人们的生活水平日益提高，生活节奏也不断加快。传统的电影购票方式，需要人们必须去到影院才可以了解到最新上映的电影的信息。这对于每天都为工作所忙碌的上班族来说是很不方便的。影院也需要安排很多的工作人员对电影、订单信息等进行管理，以维持影院的正常运行。

1.1.2 研究意义

影院购票系统与传统的电影购票相比，更加的便捷。用户足不出户就可以查看到影院最新上映的电影信息，不用再繁琐的到线下的门面去咨询，有效的提高了用户购票的满意度。影院购票系统的后台管理界面为影院的工作人员提供了很多的管理功能，在减轻影院的运营成本的同时，有效的提高了工作人员对整个影院运营的管理。

1.2 国内外文献综述

网上购票给人们的生活带来了便利，它不仅使人们可以随时随地浏览各种新上映的电影的信息、还可以让用户通过网络直接对心仪的电影票进行下单。通过对文献[1]的学习，了解到目前网上存在的购票系统，系统的逻辑清晰缜密，运行流畅。但是，仅仅是流畅的运行并不能满足人们对于优质的购票体验的追求，对于客户的购票体验的优化，还存在很大的提升空间。

文献《在线购票网站的设计与实现》[2]介绍了SpringBooot整合框架技术和MySQL查询优化这两方面。使用SpringBooot框架的好处有很多，可以节约开发时间，降低开发的成本而且是系统具有良好的扩展性。但是目前设计的系统对于订单的管理不够细致，而且作为商家的影院管理者，对于影院的一些展示公告之类的功能缺失。这些问题有待优化处理。

文献《基于Java EE的景点购票系统的设计与实现》[3]针对人工售票效率低下问题，基于B/S开发模式，采用Node.js的Express框架，结合MySQL数据库设计开发了一款影院购票系统，实现了用户通过登录用户账户对电影进行下单购票、根据自己对电影的见解对电影进行评论，管理员通过登录管理员账户可以对用户注册的基本信息进行管理操作，还可以对上映的电影的基本信息进行修改，还可以发布新的电影信息，还可以对用户对电影发表的评论进行管理，还可以对用户创建的订单进行管理等功能[4]。经过系统测试，该系统各项功能完善、性能稳定，能达到影院购票及管理需求。

文献《客票系统在线购票接口方案》[5]作者采用Java编程语言，以idea作为主要的开发工具，用Vue框架为前台开发语言，后台数据库采用Oracle公司的MySQL 数据库，经过开发设计出了一个实用、简单并且功能俱全的在线影院购票系统，该系统的前台要求实现用户的个人信息注册、用户登录，用户浏览上映电影的信息、用户创建订单等功能[6]。

文献《Set up an online booking system》[7]实现网络化的管理模式，注定会得到客户的喜爱，提高自身的竞争力。影院购票系统在增加用户的购票体验的同时也要便于管理员的日常管理与更新维护，整个系统需要简单便捷运行流畅，并具有很好的扩展性。

1.3 研究的主要内容和方法

1.3.1 研究内容

影院购票系统的系统角色有管理员和用户，主要实现的是系统功能用户登录、浏览电影信息、购票、支付、对电影进行评价、管理员登录、订单管理、影院信息管理。本文的研究内容如下：

（1）用户信息管理及登录，用户可以通过注册获得自己的用户账户，并且可以对自己的用户信息进行修改，用户通过输入自己设置的用户名和密码进行登录操作。

（2）用户浏览影院信息进行管理，用户通过登录自己注册的用户进入到影院购票系统的前台页面，在影院购票系统的前台页面，账户可以查看到影院的电影信息，并根据自己的需求就行浏览挑选。

（3）我的订单管理，用户挑选到自己心仪的电影票后，对电影票进行下单。管理员可以通过管理员账户在影院购票系统的后台界面对用户提交的订单进行管理操作。

（4）影院信息管理，管理员通过登录管理员账户获取修改影院信息的权限，从而对影院信息进行管理和维护操作。

1.3.2 研究方法

本文主要使用调查法、文献研究法等方法进行研究，具体如下：

（1）调查法：通过对喜欢来电影院看电影的人进行发放调查问卷和随机现场采访的方式，了解他们对电影购票的看法和想法，以完成搭建影院购票系统所需要的数据。再对这些收集到的数据进行归纳、分析、总结。根据总结确定影院购票系统的功能设计思路。

（2）文献研究法：通过查阅与影院购票系统设计相关的文献资料，对查阅的文献资料进行归纳、总结，从中获取到国内外关于定义购票系统的设计的研究现状和现阶段还存在的问题，在这个基础上进行系统的需求分析。

1.4 所用技术介绍

1.4.1 Java编程语言介绍

Java对比其他语言来说都有显著的优势。

1. 丰富的框架，Java语言具有丰富的Java框架，这大大的提升了我们的开发效率。
2. 面向对象，Java语言是一种面向对象的语言，它通过使用一些基础的方法就可以把规定的任务做完，只需要对Java语言的基本概念有所理解，就可以使用Java写出适合开发任务的相关应用程序。
3. 跨平台性，Java语言可以在不同的平台进行部署和执行，源代码和二进制平台的独立性是Java在各方面成功的所在。

1.4.2 SpringBooot介绍

Spring Boot 是一个为了简化 Spring 应用程序的创建和开发而设计的开源框架。它基于 Java 平台，使用 Spring 框架作为基础，并封装了大量的功能和常用配置，以便开发人员可以更快速、更便捷地搭建和部署应用程序。

Spring Boot 的设计思想是约定优于配置，通过自动配置、自动装配和默认配置来减少开发人员的工作量。它提供了一套强大的开发工具和开箱即用的功能，使得开发者可以将更多的精力放在业务逻辑上，而不是繁琐的配置上。

Spring Boot 具有以下特点和优势：

简化配置：Spring Boot 提供了自动配置的机制，根据项目中引入的依赖和默认规则，自动配置 Spring 应用程序的各种组件和功能，无需手动编写大量的配置文件。

微服务支持：Spring Boot 提供了对微服务架构的良好支持。通过 Spring Boot 可以快速构建和部署微服务应用，并提供诸如服务发现、负载均衡、断路器等功能。

内嵌容器：Spring Boot 默认集成了 Tomcat、Jetty 或 Undertow 等内嵌容器，可以直接将应用程序打包为可执行的 JAR 文件，并以独立的方式运行，无需外部容器的支持。

健康监测：Spring Boot 提供了健康监测的功能，可以检测应用程序的运行状态并提供相关的管理端点，方便运维人员进行监控和管理。

准生产特性：Spring Boot 提供了一系列的特性，如安全、日志、数据访问等，可以帮助开发者快速地将应用程序推向准生产环境。

生态系统：Spring Boot 是 Spring 生态系统的一部分，与其他 Spring 模块无缝集成，可以享受到 Spring 框架强大的功能和丰富的第三方库的支持。

1.4.3 MySQL数据库介绍

MySQL是一个安全的，跨平台的，有效的，并且与主流的程序语言，如PHP,Java，有密切的联系。

MySQL数据库管理系统的特点：

它是C和C++语言编写的、支持多个操作系统、支持多线程、为多种编程语言提供API、优化SQL算法提高了查询速度以及提供用于管理和检查MySQL数据库是数据库的管理工具。MySQL数据库可以把数据储存在不同的表中[9]。因此它的灵活性强，速度快强。它具有两种模式：商业版跟社区版，一般新手都会选择社区版来创建自己的数据库系统。MySQL数据库深受中小型公司的青睐，因为MySQL数据库具有占用内存小，运行速度快，开发成本低，且源码开放等诸多优点[10]。

2 系统分析

2.1 可行性分析

2.1.1 经济可行性

目前采用计算机进行具有：成本低，实用性强，性能增强，运行效率高等优势。在降低了人工成本的同时，也减轻了管理人员的劳动负担。在保障了文档的保密性的同时，使得工作的效率得到了明显的提示。

在现有条件下，开发该系统所需要的设备只用少量的资金就可以实现，可以最大限度地降低成本和支出，因此在经济上是可行的。

2.1.2 技术可行性

本系统采用Java语言，使用SpringBoot框架以及数据库MySQL共同开发，后台数据库采用JDBC方式进行连接，完成对数据的增加、查询、修改、删除。MySQL灵活易维护。因此使用SpringBoot＋MySQL成为轻平台开发的最佳技术。

2.1.3 社会因素可行性

本系统为独立开发，没有抄袭任何现有的相关系统，所以在法律上不存在任何的违法、侵犯知识产权等行为。本系统也没有违背国家、地方政府的方针和导向不属于行业垄断范围，没有产生任何的不公平竞争。

2.2 需求分析

2.2.1 功能需求分析

本系统主要是实现用户网上订购电影票的需求，其中，对于用户的信息管理及订单的管理尤为重要[11]。用户通过登录自己的用户账户进入到系统的前台页面对电影信息进行浏览，并对心仪的电影进行下单支付，用户根据自己的观影体验可以对电影发表评价[12]。管理员通过登录管理员账户登录到系统的后台界面，在系统的后台对电影的信息、用户的信息、电影详情介绍信息、影院公告信息和订单的信息进行管理操作[13]。因此系统应实现以下功能：

（1）用户需要先注册用户账户；

（2）用户使用自己注册的账户，并输入正确的密码进行登录。

（3）用户登录到系统之后，可以浏览到影院的电影信息。

（4）用户能对心仪的电影票进行挑选和购买。

（5）用户查看电影具体详细信息

（6）管理员通过管理员账户可以对影院的基础信息进行管理操作。

（7）管理员通过管理员账户可以对订单信息进行管理操作。

（8）管理员通过管理员账户可以对影片详情介绍信息进行管理操作。

（9）管理员通过管理员账户可以对影厅信息管理

用户用例图如下：

（1）用户用例图，如图2-1所示。



图2-1 用户用例图

（2）管理员用例图，如图2-2所示。



图2-2 管理员用例图

2.2.2 性能需求分析

（1）系统的各个界面要清楚简洁，易于用户上手使用。

（2）订单界面要简明，让用户能够清楚的看到各种信息。

（3）管理电影界面需简明，以便于管理员对于电影信息的管理。

（4）通过使用该系统，用户挑选和购买电影票更便捷，影院管理会更轻松。

3 系统设计

3.1 系统结构设计

影院购票管理系统的基本系统结构可以分为：用户的信息管理模块、管理员的信息管理模块、订单信息的管理模块及电影信息的管理模块。如图3-1所示。



图3-1 影院购票管理系统功能图

3.2 功能模块设计

影院购票系统的几个主要的功能模块：登录模块、订单管理模块、电影信息的管理模块等功能模块组成，主要的目的是为了为用户购票提供便捷，同时也方便了工作人员的对订单和电影信息的管理。

（1）登录模块

用户打开系统的界面之后通过输入正确的用户名和与之匹配的正确的密码进行用户登录操作，这个时候，系统会对用户输入的用户名和密码进行判断，判断该用户所输入的用户名和密码是否正确，如果该用户所输入的用户名和密码正确用户就可以成功的登录到影院购票系统，反之登录失败需重新输入用户名和密码。如图3-2所示。



图3-2 登录程序流程图

（2）订单管理模块

订单管理模块主要是对订单的填写、订单的修改和订单的删除等操作，用户可根据自己的需求填写生成订单，还可以对订单进行修改和删除操作。管理员可以通过管理员账户对订单进行查询，以进行订单的管理操作。如图3-3所示。



图3-3 订单管理程序流程图

（3）电影信息的管理功能模块

电影信息的管理功能是用来管理电影的基本信息的，包括：电影名，宣传海报，影票价格等。由管理员通过登录到管理员账户进行统一的管理。用户可用过登录用户账户对影院出示的电影进行挑选和购票操作。

（4）搜索模块

用户通过输入正确的用户名和密码登录到系统中，使用自己创建的普通用户进行电影票的浏览和购买，用户在浏览电影的时候可以通过搜索功能根据需求在系统中的电影信息进行搜索。用户只需要输入关键字，然后点击搜索，系统就会及时的显示出根据用户输入的关键字所搜索到的电影信息，并列举出来。如图3-4所示。



图3-4 搜索程序流程图

（5）用户管理功能模块

用户首次登录需要注册自己的账户，设置账户密码，填写个人信息，即可创建个人账户。并可以用过用户中心，对自己的账户信息进行修改。

（6）管理员的管理功能模块

管理员可通过输入正确的管理员用户名和密码登录到影院购票系统的后台，在影院购票系统的管理后台对管理员的账户基本信息进行添加、查询、删除、修改等功能进行操作，以便于在人员调整的时候对管理员账户进行管理操作。

3.3 数据库设计

3.3.1 数据库概念设计

当前，数据库技术是一种能够有效地管理真实世界中巨大的数据资源的手段[14]。在软件项目中的数据库设计指的是以系统项目的需求分析为基础，以与软件开发标准相一致的数据库模型为基础，建立一个能够有效地保存数据的系统数据库，以此来达到对用户数据的需要和处理标准[15]。在设计影院购票系统的数据库的时候，需要非常注意数据库的安全性、稳定性和恢复性，为了使数据库稳定可靠，需要以需求分析为依据，展开详细的数据库设计。数据库的E-R图如图3-5所示。



图3-5 数据库E-R图

3.3.2 数据库逻辑设计

（1）管理员的基本信息表，该表存储了管理员用户的基本信息，如表3-1所示。

表3-1 管理员的基本信息表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **字段的类型** | **字段的字符大小** | **字段的文字说明** |
| user\_id | int | 20 | 编号 |
| user\_name | varchar | 50 | 用户名 |
| password | int | 50 | 密码 |
| user\_picture | varchar | 40 | 头像 |
| phone\_number | int | 11 | 手机号 |

（2）用户的基本信息表，该表存储了用户的基本信息，如表3-2所示。

表3-2 用户的基本信息表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **字段的类型** | **字段的字符大小** | **字段的文字说明** |
| user\_id | int | 100 | 编号 |
| user­\_name | varchar | 50 | 用户名 |
| password | int | 50 | 密码 |
| user\_picture | varchar | 40 | 头像 |
| sex | varchar | 2 | 性别 |
| birthday | int | 10 | 生日 |
| phone\_number | int | 11 | 手机号 |

（3）电影信息表，储存电影信息如表3-3所示。

表3-3 电影信息表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **字段的类型** | **字段的字符大小** | **字段的文字说明** |
| movie\_id | int | 100 | 编号 |
| movie\_name | varchar | 50 | 名称 |
| release\_date | varchar | 60 | 上映时间 |
| movie\_introduction | varchar | 1000 | 简介 |

（4）订单的基本信息表，该表存储了所生成的订单的基本信息，如表3-4所示。

表3-4 订单的基本信息表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **字段的类型** | **字段的字符大小** | **字段的文字说明** |
| bill\_id | int | 100 | 编号 |
| seats | int | 100 | 座位号 |
| cancel\_time | datetime | 100 | 取消时间 |
| create\_time | int | 100 | 创建时间 |
| pay\_state | varchar | 5 | 支付状态 |

4 系统实现

4.1 系统前台界面的实现

影院购票系统的前台界面设计主要包含了浏览页面、用户注册页面、用户登录页面、用户信息管理页面、订单页面、电影详情介绍功能的设计与实现。

4.1.1 用户注册登录功能的实现

（1）系统界面简约便捷，用户可以轻松的上手。点击“用户注册”即可开始进行用户账户的注册。在这个界面用户可以在对应的输入框中输入自己想要使用的账户的用户名和想要设置的密码，然后填写用户的基本信息：用户的真实姓名和用户的联系方式，完成之后点击“确认注册”按钮即可完成用户注册操作。如图4-1所示。



图4-1 用户注册界面

（2）用户在完成注册之后便可用自己注册的账户进行登录，登录到系统之后就可以浏览到影院购票系统前台界面中的电影信息内容。如图4-2所示。



图4-2 用户登录界面

4.1.2 用户订单管理功能的实现

（1）用户通过浏览电影信息选择到自己心仪的电影票，并购买，在购买 完成后，个人中心中的我的订单查询详细信息前端代码，使用的是Vue框架和ElementUI组件库。代码中包含了面包屑导航、搜索栏、订单列表、分页等功能。

在模板部分，使用了<template></template>标签包裹需要渲染的页面内容。使用了ElementUI的组件，如<el-breadcrumb>面包屑导航、<el-card>卡片、<el-row>和<el-col>栅格布局等。列表使用了<el-table>表格组件展示订单信息，分页使用了<el-pagination>分页组件。

在脚本部分，定义了一些数据和方法。数据部分包括了查询条件、订单列表、总数、多选框选择、修改对话框的显示和表单数据等。方法部分包括了获取订单列表、分页、多选框选择、显示修改对话框、关闭修改对话框、修改订单信息等。如图4-3所示。



图4-3 加入购物车界面

（2）用户对下单完成之后系统会生成订单。以方便用户查看订单详情。这个组件使用了Element UI库来实现步骤条（el-steps）和按钮（el-button）等组件。在data中定义了一些变量用于存储订单的信息，例如订单编号（billId）、订单详情（billInfo）、订单座位（billSeats）、支付状态（payState）、取消状态（cancelState）等等。在created钩子函数中调用了getBillInfo方法来获取订单的详细信息，并将信息赋值给相应的变量。mounted钩子函数用于启动计时器，计算剩余支付时间（minutes和seconds），并通过add方法来进行倒计时。如图4-4所示。

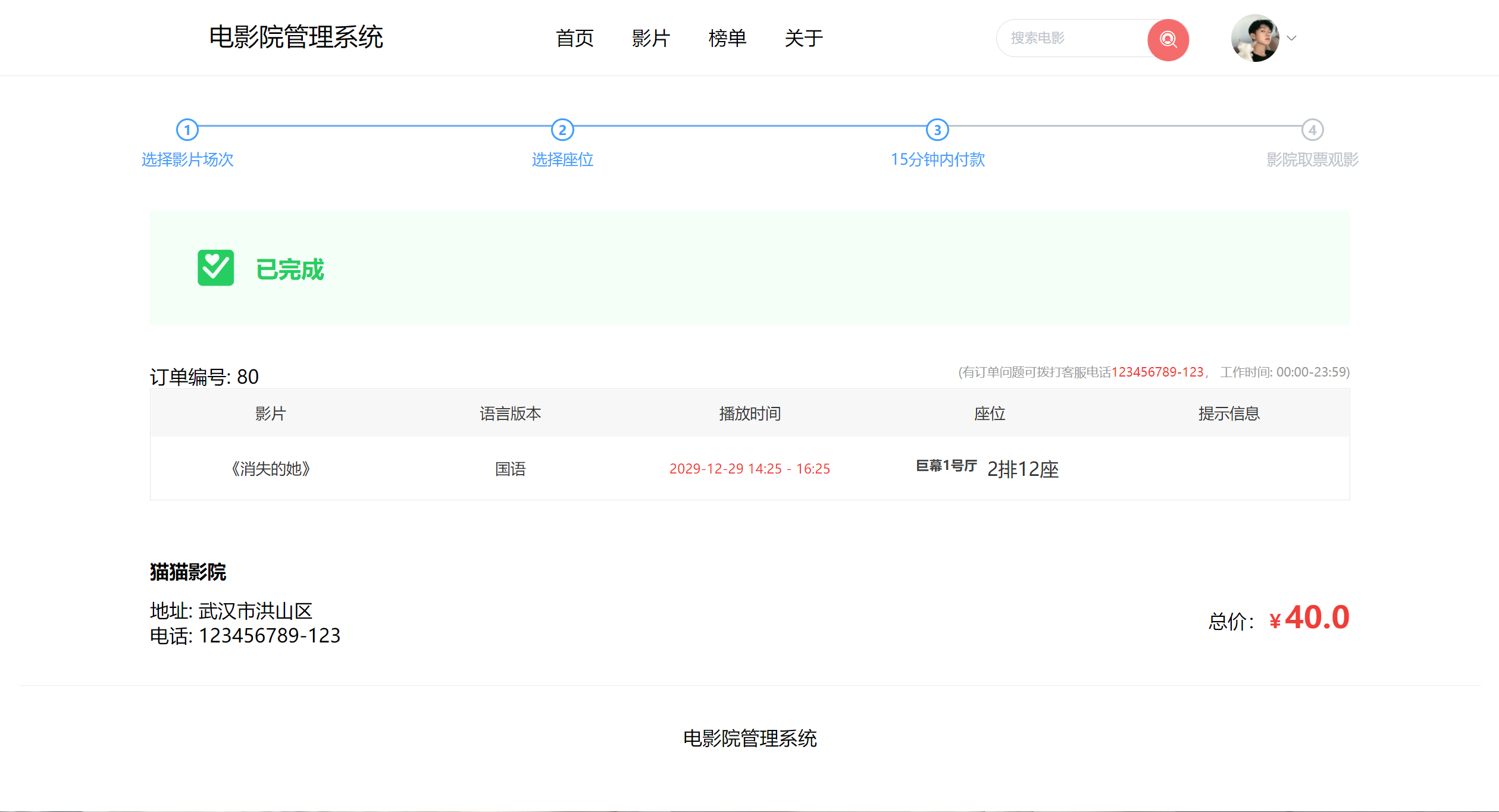


图4-4 生成订单界面

4.1.3 用户信息管理功能的实现

用户可以通过点击“个人中心”按钮进入个人中心界面，在该界面，用户可以根据自己的需求对自己的账户信息进行编辑。如图4-5所示。

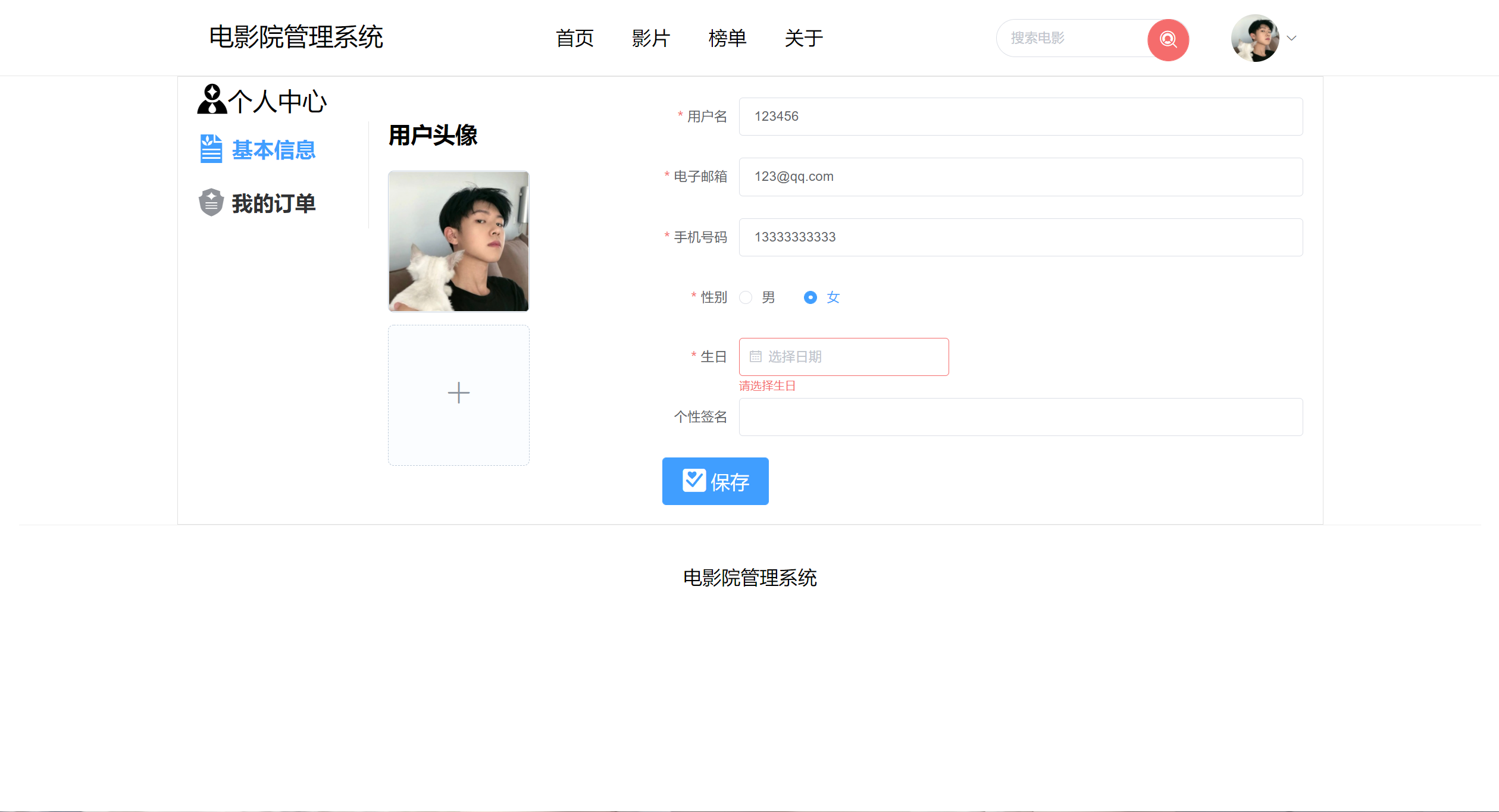


图4-5 用户信息管理界面

4.1.4 用户账户密码管理功能的实现

用户可以通过个人中心，可以对用户的账户密码进行更改操作，以保障账户的安全。如图4-6所示。

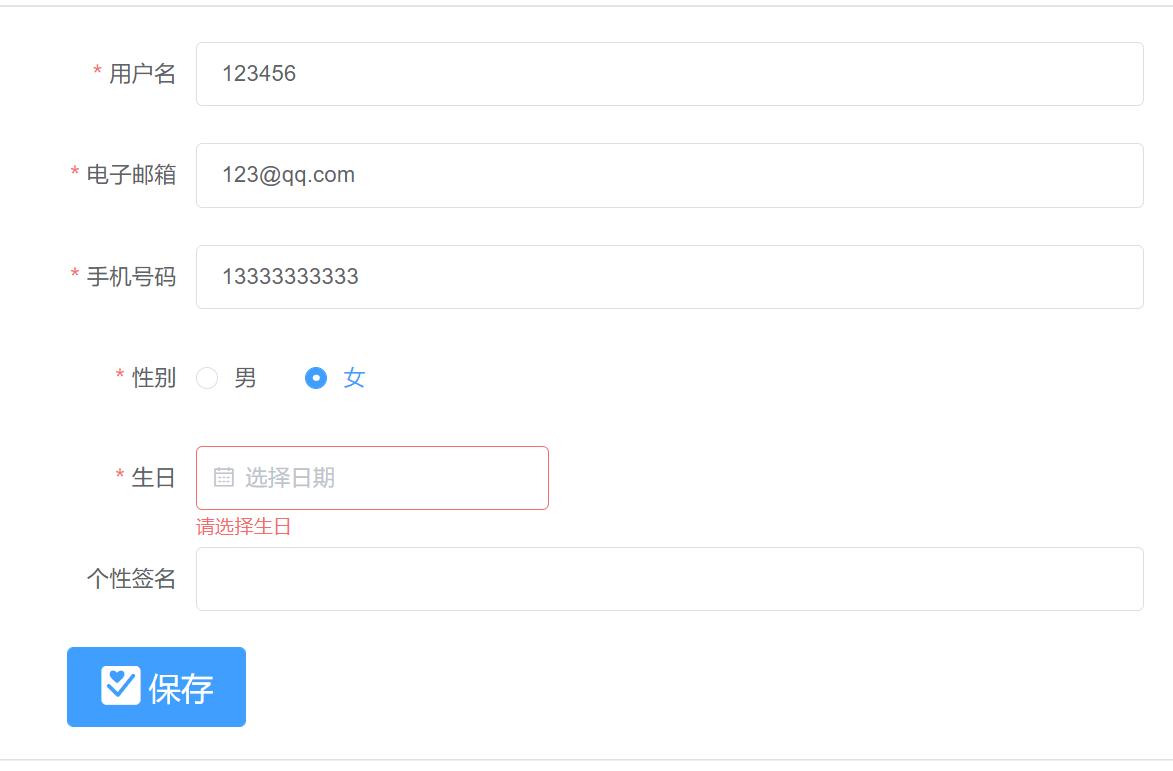


图4-6 修改用户账户密码界面

4.1.5 用户电影详情介绍功能的实现

用户浏览电影之后，可以根据电影查看详情信息代码主要包括模板(template)、脚本(script)和样式(style)三个部分。

模板部分是用来定义组件的结构和布局，其中使用了一些Vue组件和Element UI组件，如el-button、el-tabs、el-tab-pane和el-image等。模板中的数据绑定使用了双大括号{{}}，例如{{movieInfo.movieName}}用来显示电影的名称。

脚本部分是组件的逻辑部分，定义了组件的名称、引入的组件和依赖，以及一些方法和数据。在created()方法中，通过发送请求获取电影相关信息，并将获取到的数据存储到movieInfo中。该部分还定义了一些点击事件方法，如toChooseSession()用于跳转到购票页面。

样式部分是用来定义组件的样式，其中使用了一些CSS属性和选择器来设置组件的布局、颜色和字体等样式。

总体而言，该组件是一个电影信息页面组件，用于显示电影的基本信息、剧情简介和图集等内容，用户可以通过点击按钮进行相关操作，比如跳转到购票页面。。如图4-7所示。



图4-7 电影详情介绍界面

4.2 系统后台界面的实现

影院购票系统的后台界面设计包含了：影院信息管理、影片管理、影厅管理、订单管理、用户管理、角色权限管理等功能。

4.2.1 管理员登录功能的实现

管理员通过输入正确的管理员账户的用户名和密码后，进入到系统后台管理页面，对影院购票系统的后台进行管理操作。如图4-8所示。



图4-8 管理员登录界面

4.2.2 影院信息管理功能的实现

页面上有一个面包屑导航区域，展示当前所在的位置。然后是一个卡片视图，显示影院的具体信息。影院信息包括影院名称、地址、电话、营业时间、拥有的影厅类型和影院图片等。

影院图片部分采用了弹出框的方式显示，点击图片弹出框会显示更大的图片。

页面下方有一个修改影院的对话框，点击"修改影院信息"按钮可以打开对话框进行影院信息的修改。在对话框中可以修改影院名称、地址、电话、营业时间、影厅类型和影院图片等信息。在影厅类型部分，可以通过输入框输入新的影厅类别名称，并点击"添加"按钮进行添加，已添加的影厅类别会以标签的形式显示在下方。

在影院图片部分，可以通过上传文件方式添加新图片，已添加的图片会以卡片的形式展示。同时，可以点击图片进行预览，并可以删除已添加的图片。

在修改影院信息完成后，点击"确定"按钮可以保存修改，并将修改结果发送给后端进行更新。如图4-9所示。



图4-9 管理员修改密码界面

4.2.3 影片信息管理功能的实现

代码中使用了el-breadcrumb组件来展示面包屑导航，el-card组件展示电影信息列表，el-table组件展示电影信息表格，el-pagination组件展示分页器，el-dialog组件展示对话框，el-form组件展示表单，el-input、el-select、el-date-picker等组件用来输入和选择数据，el-button组件用来触发事件，el-upload组件用来上传文件。

整个页面包含了面包屑导航区、搜索区、电影信息列表、分页区域以及添加电影对话框和修改电影信息对话框。

在电影信息列表中，使用el-table-column组件定义了表格的每一列，使用了插槽来自定义操作列中的按钮和事件。

在添加电影对话框和修改电影信息对话框中，使用el-form组件来实现表单验证和数据绑定，el-input组件、el-select组件和el-date-picker组件用来输入和选择电影信息，el-upload组件用来上传电影封面和图集。

整个代码使用了Vue.js的双向数据绑定和事件绑定，通过调用事件和提交表单来实现页面的功能。如图4-10所示。

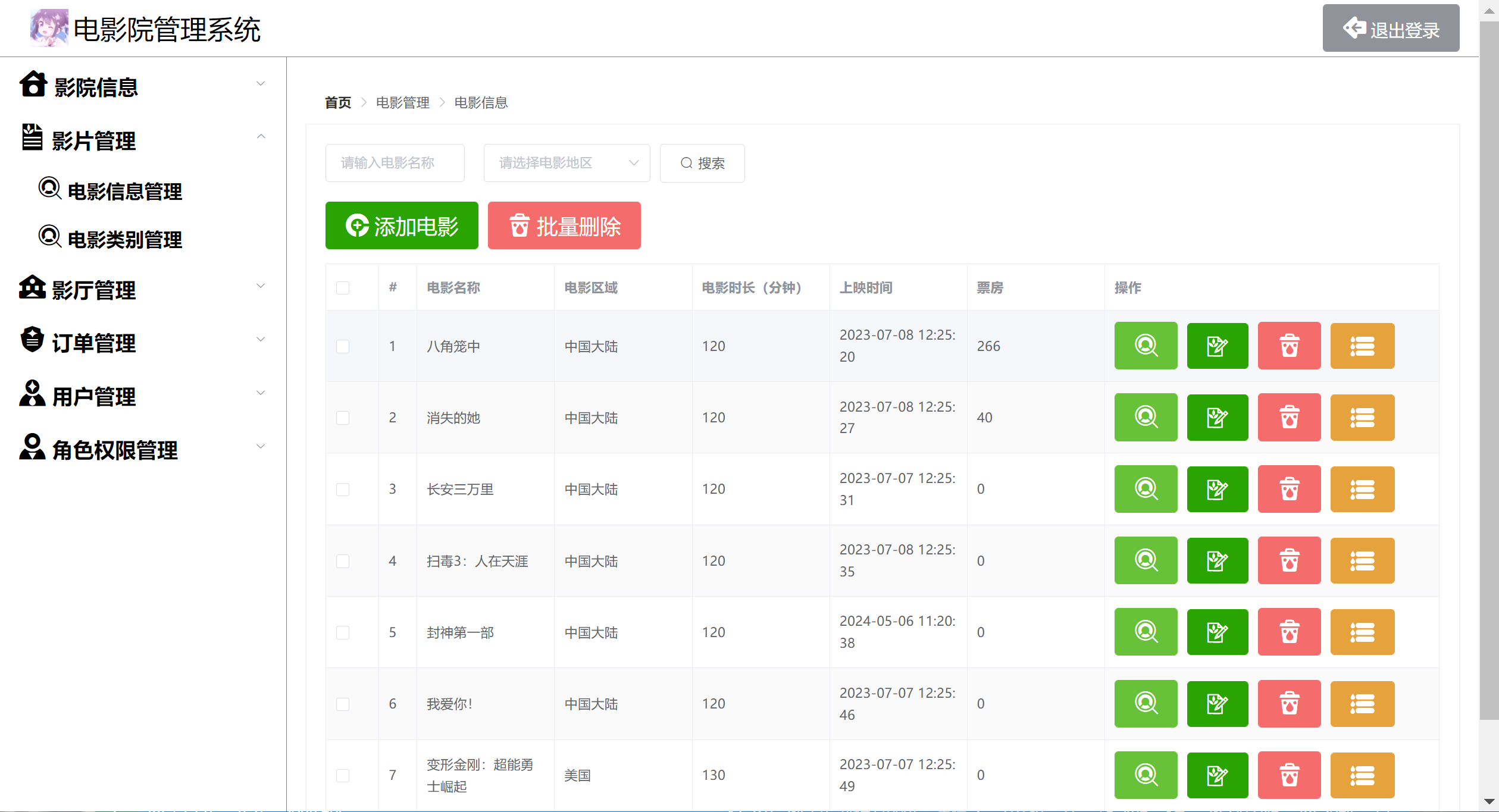


图4-10 添加新的电影信息界面

在这里，管理员可以对已上映但是存在信息错误的电影的基本信息进行修改。如图4-11所示。

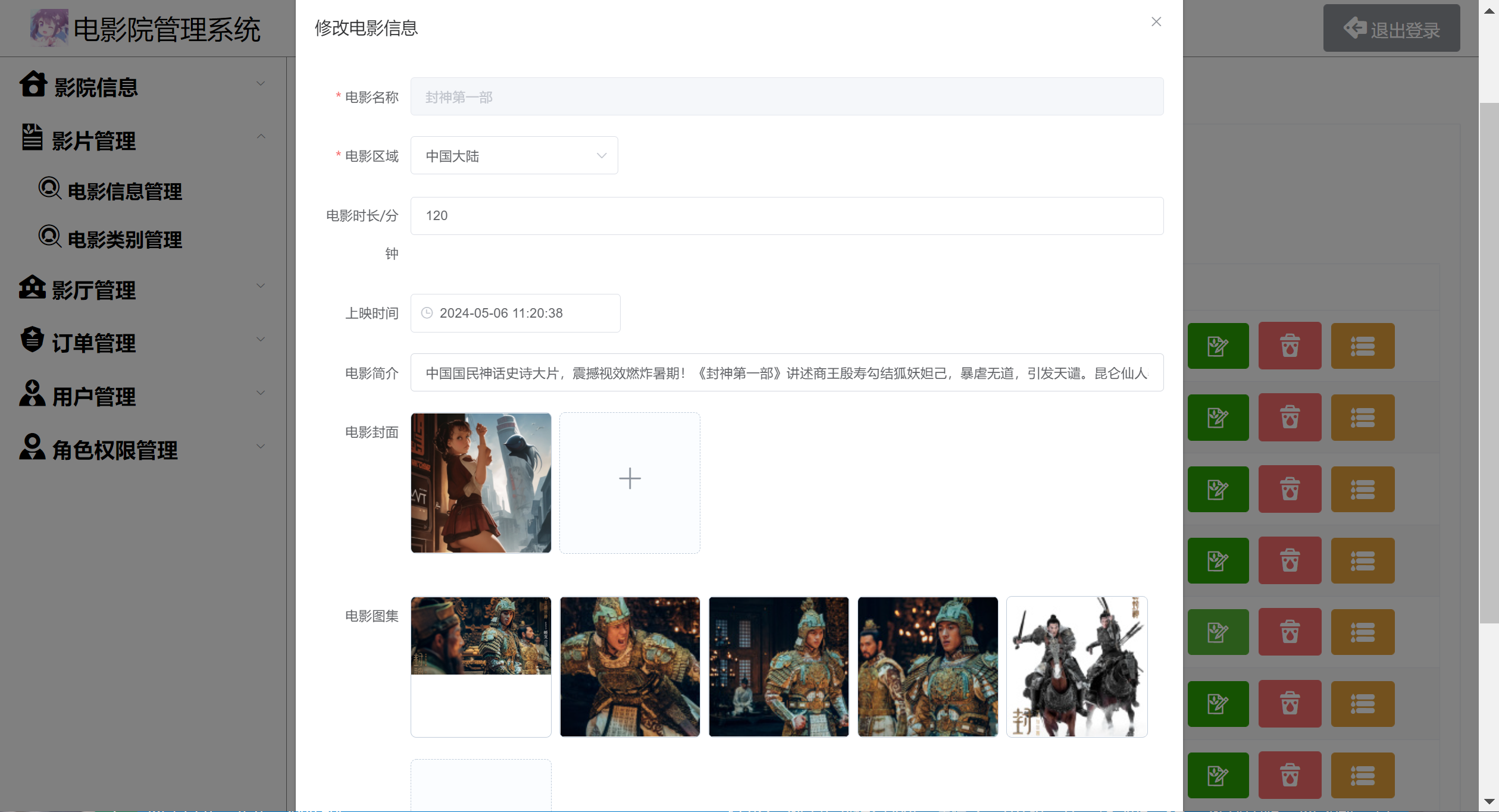


图4-11 修改电影信息界面

4.2.4 订单管理功能的实现

管理员可通过管理员账户进入到系统之后，通过系统的后台管理功能对用户所创建的订单进行统一的管理操作。模板部分（<template>标签中）包含了页面的布局结构，包括面包屑导航和卡片视图。在面包屑导航中，使用了el-breadcrumb和el-breadcrumb-item组件来实现导航的展示和点击事件。在卡片视图中，使用了el-card、el-row和el-col组件来实现布局，其中包含了一些输入框、下拉框、日期选择器和按钮。

脚本部分（<script>标签中）定义了页面的数据和方法。数据部分包括了一些查询条件、订单列表、总条数和选中的订单信息。方法部分定义了一些操作订单的方法，如获取订单列表、分页、多选、显示编辑对话框等。

在模板中使用了el-table组件来展示订单列表，通过设置:data绑定billList变量来显示订单数据，通过设置@selection-change来监听多选框的变化。

在表格的操作列中，使用了自定义的slot来显示操作按钮，当点击按钮时，会触发对应的方法，如显示编辑对话框和删除订单。。如图4-12所示。



图4-12 订单管理界面

4.2.5 用户管理管理功能的实现

使用了 Vue.js 和 Element UI 组件库。

页面上有一个面包屑导航区域，用来显示当前页面所处的位置。

接下来是一个卡片视图，包括了一个搜索框和三个输入框，用来输入用户名、邮箱和手机号码进行搜索。还有一个下拉框用来选择性别。最上方有三个按钮，分别用来执行索、添加用户和批量删除操作。

用户信息以表格的形式显示，包括选中框、序号、用户名、电子邮箱、手机号码、性别和角色等信息。操作栏包括修改用户信息和删除用户两个按钮。

最下方是一个分页区域，可以切换每页显示的数量和页码。

页面还包括两个对话框，分别用于添加和修改用户信息。添加用户对话框中包含了用户名、密码、电子邮箱、手机号码、性别和权限等输入框和下拉框。还有一个功能用于上传用户头像。修改用户对话框与添加用户对话框类似。

整个页面的样式使用了一个全局CSS文件。

该页面的功能主要是进行用户信息的管理，包括搜索、添加、修改和删除用户等操作。如图4-13所示。

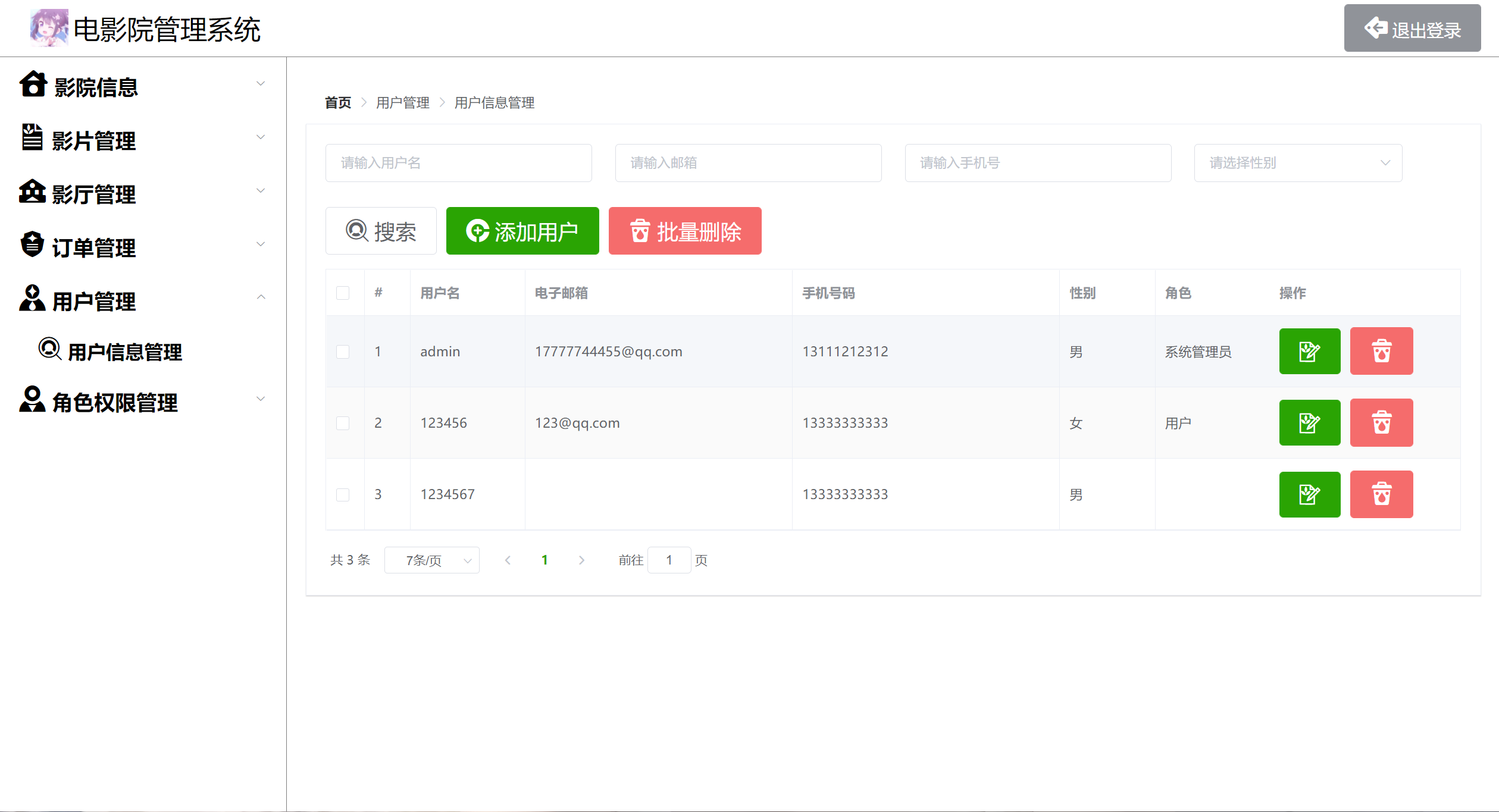


图4-13 电影详情介绍管理界面

4.2.6 影厅管理功能的实现

该代码是一个Vue组件，主要是一个影厅信息管理的页面模板。页面由若干个元素组件组成，包括面包屑导航、卡片视图、影厅分类列表、分页区域、添加影厅对话框、修改影厅对话框和安排座位对话框。

在模板中，使用了Element UI框架提供的组件，比如el-breadcrumb、el-card、el-row、el-col等。使用这些组件可以快速构建出页面的结构和样式。

代码中使用了一些Vue的特性，比如数据绑定（使用v-model指令）、条件渲染（使用v-if指令和JavaScript表达式）、循环渲染（使用v-for指令）、事件处理（使用@click指令）等。

整个页面的逻辑由Vue组件data中的数据和methods中的方法控制。在data中定义了数据源和控制对话框显示隐藏的变量，在methods中定义了各种操作函数，比如获取影厅列表、添加影厅、删除影厅等。如图4-14所示。

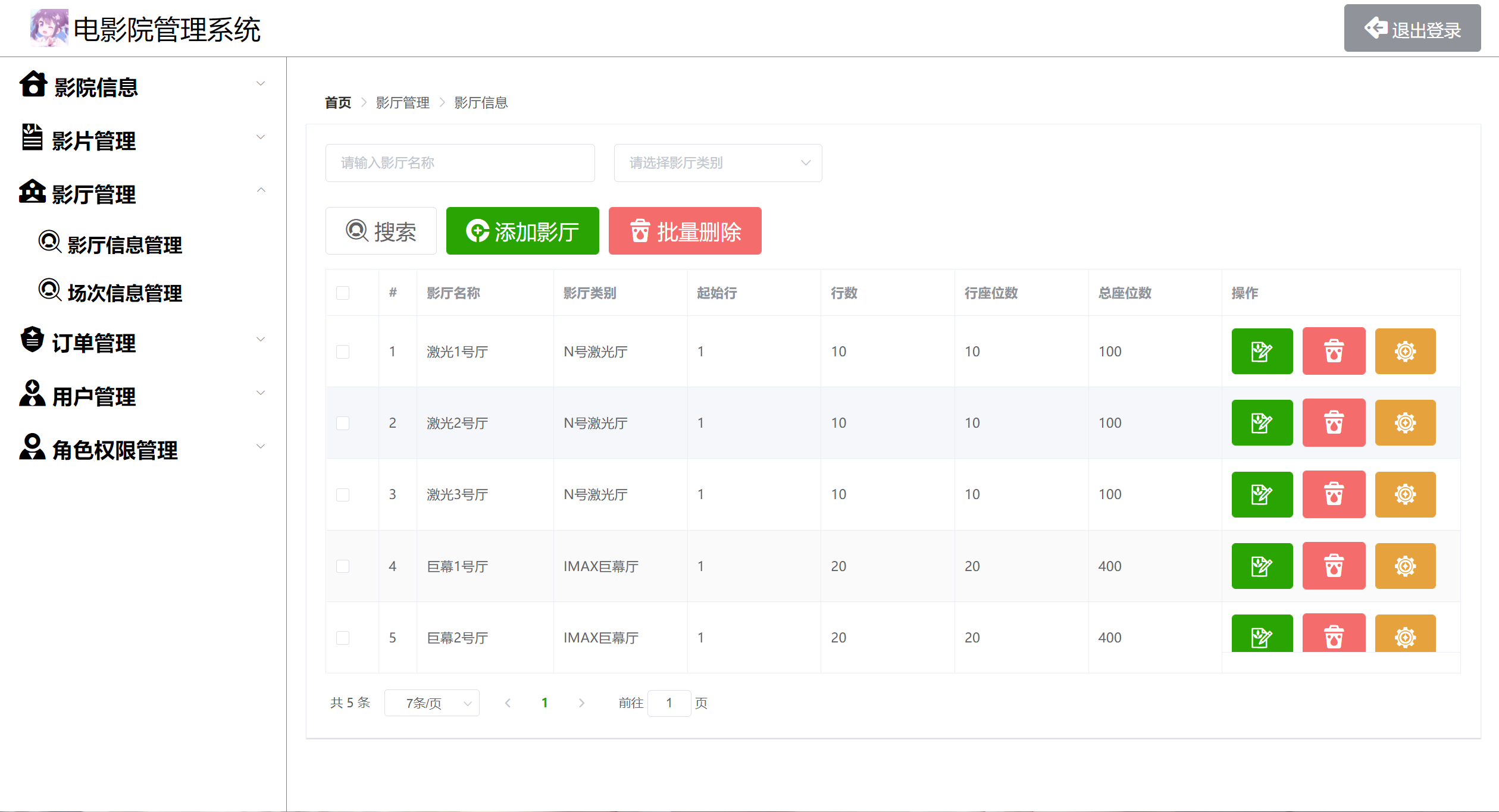


图4-14 影院公告管理界面

5 系统测试

5.1 测试目的及方法

系统测试是系统开发过程中的一个十分重要的环节。其重要意义在于，其是确保系统后期的使用可以稳定高效的运行，这个环节是对系统开发的前面所有环境的一个审查，是为了让系统能够正式的投入到实际的使用的重要的的一个环节。

在此基础上，提出了一种基于计算机技术的计算机辅助设计方法，并将其应用于计算机辅助设计中。系统的缺陷主要有：函数缺陷，系统缺陷，程序缺陷，数据缺陷，代码缺陷等。

在对该系统的测试中，我们采用了多种系统测试方法，对系统的主页、各功能模块及页面、数据库操作、程序代码和总体功能进行了具体的测试，确保了系统的质量、可靠性和可维护性。

对软件进行测试，就是要找出软件中存在的问题，然后进行修正。对于一个关于测试的不正确看法应当被摒弃，应当适当地去选择一些适当的案例，并且在尽量减少的情况下，找到尽量多的程序缺陷。

本文采用功能测试对影院购票系统的前台界面和后台界面分别进行测试。

5.2 测试用例

5.2.1 系统前台功能性测试

系统前台功能测试主要测试影院购票管理系统的用户登录、下单等功能是否可以正常运行，通过输入不同的数据，查看实际结构是否达到预期的结果。系统前台功能性测试表如表5-1所示。

表5-1 系统前台功能性测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| **测试内容** | **测试操作描述** | **期望结果** | **实际结果** |
| 用户登录 | 1.录入正确的用户名：lihua、密码：123456 | 登录成功 | 登录成功 |
| 2.录入正确的用户：lihua、不正确的密码：145263 | 登录失败 | 登录失败 |
| 3.录入不正确的用户：xiaoli、正确的密码：123456 | 登录失败 | 登录失败 |
| 下单 | 1.电影票点击下订单并支付 | 正常下单 | 正常下单 |

5.2.2 系统后台功能性测试

系统后台功能测试主要测试影院购票管理系统的管理员登录、电影信息管理、订单管理、影厅管理等功能是否可以正常运行，通过输入不同的数据，查看实际结构是否达到预期的结果。系统前台功能性测试表如表5-2所示。

表5-2 系统后台功能性测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| **测试内容** | **测试操作描述** | **期望结果** | **实际结果** |
| 管理员登录 | 1.录入正确的用户名：admin、密码：admin | 登录成功 | 登录成功 |
| 2.录入正确的用户：admin、不正确的密码：123456 | 登录失败 | 登录失败 |
| 3.录入不正确的用户：lihua、正确的密码：admin | 登录失败 | 登录失败 |
| 电影信息管理 | 1.修改已上映的电影结束时间 | 修改成功 | 修改成功 |
| 2.发布新的电影信息 | 发布成功 | 发布成功 |
| 3.删除电影信息 | 删除成功 | 删除成功 |
| 订单管理 | 1.修改订单信息支付状态 | 修改成功 | 修改成功 |
| 2.查询订单 | 查询成功 | 查询成功 |
| 3.删除订单 | 删除成功 | 删除成功 |

5.3 测试结论

系统测试之后，针对在系统调试过程中遇到的一些问题，我们已进行了调试与维修，此期旨在确保电影预订系统能够稳定、稳定地运转，并且能够持续改进与改进，从而最大限度地将其功能发挥出来。在系统的整体运作中，一直都是对系统的维护与管理，主要是对系统的维护、代码的维护、数据的备份与恢复、硬件设备的维护。在进行操作的时候，也要跟着环境的改变，以各种的需要及变化为依据，对系统展开相应的改进，从而让系统的功能变得更加完美。

结论

影院购票系统采用Java作为开发语言，MySQL作为后台数据库。系统分为前台购物界面和后台管理界面，系统的前台界面为用户提供了浏览，购票，管理等功能，系统后台界面为用户提供了丰富的管理功能。该系统为用户提供更便捷的购票方式，也提升了影院管理人员对于影院运营的管理。简约的系统界面设计，让用户能够轻松的上手使用该系统，便捷的查询功能可以让用户凭借关键字快速的找到自己心仪的电影信息。简约的后台管理界面，使得管理员可以对整个影院购票系统进行统一管理。

在本文设计的影院购票系统中，在对其进行的需要考虑和对其进行的数据库的设计方面，存在着一些缺陷，因此，与目前的企业使用的管理系统相比，仍存在着一定的差距。希望以后能对这个系统进行进一步的优化完善，让它成为一个更好的系统。

参考文献

1. 王良升,刘小英.电影院购票系统的设计与实现[J].电脑编程技巧与维护,2021(01):54-56.
2. 李天翔,田林琳.在线购票网站的设计与实现[J].湖北农机化,2019(14):81.
3. 孙勇.基于JavaEE的景点购票系统的设计与实现[J].科技风,2020(08):116.
4. 黄向东,解艳君.企业差旅购票系统设计与应用[J].中国管理信息化,2020,23(01):187-189.
5. 罗彤.客票系统在线购票接口方案[J].铁路通信信号工程技术,2018,15(09):51-54.
6. 陈万烤. 航空购票服务器爬虫检测技术研究[D].杭州电子科技大学,2018.
7. Anonymous. Set up an online booking system[J]. Computer Actve,2022(643).
8. 蔡明山,刘磊,李俊峰.影院购票管理系统设计[J].榆林学院学报,2018,28(06):98-100.
9. 周程帆,王贵鑫.基于Android平台的影院票务管理系统的设计与实现[J].科学技术创新,2019(32):62-64.
10. 王红爱. 铁路客运退票风险管理研究[D].中国铁道科学研究院,2021.
11. 苑修坦. 铁路客运服务投诉系统数据库建设及政府监管建议研究[D].大连交通大学,2018.
12. 陈舟劢. 贵州旅游景点智能推荐系统的设计与实现[D].贵州大学,2021.
13. 周定贤. 名芝雪糕公司订单管理系统优化方案研究[D].吉林大学,2022.
14. Tiwari Ashish,Garg Ritu. Reservation System for Cloud Computing Resources(RSCC): Immediate Reservation of the Computing Mechanism[J]. InternationalJournal of Cloud Applications and Computing (IJCAC),2022,12(1).4
15. Carlan Valentin,Hintjens Joost,Stuer Julie,De Vylder Dirk,Vanelslander Thierry. Why do slot booking systems still generate time-losses and is there a solution forit?[J]. Research in Transportation Business & Management,2022,44.

致谢

大学的学习已经接近尾声，回望过去全部都是无比珍贵的回忆，每次都是结尾时才发现时间过得如此之快，虽然在大学里我学的很多，但还不够多，我还需要继续学习，在未来，我希望我可以继续努力奋斗，不让自己后悔。

我的毕业设计能够顺利进行离不开我的毕业设计指导老师对我的指导。在此我非常感谢指导老师能够在繁忙的教学工作中抽出时间来对我的毕业设计进行审查和修改。以及所有教过我的老师们，感谢老师的付出，相信未来学生一定会在某段时间不经意的想起自己的大学生活，感谢同学们在大学期间给予我的帮助，因为认识了大家，才会觉得这四年的大学时光是值得的。

同时，也非常感谢同学之间的鼓励帮助，以及家里长辈的支持心怀。学习不仅仅是在学校的事情，未来我也要当一名学生，无时无刻的学习新的技术，而这种信心和信念也需要身边的人鼓励才能坚持下去，未来之大，一眼望不下。学海无涯，轻舟载我行。最后，感谢老师和同学的帮助和鼓励，谢谢大家。