摘要

随着我国电影行业的迅猛发展，电影院的数量以及观影人数也呈直线上升趋势，这就引发了一系列问题，如电影院管理成本提高，用户购票难、购票效率低等问题。为了降低电影院的管理成本，提升管理效率，也为了方便广大用户购买电影票，影院自助购票系统应运而生。

本文首先介绍了在线电影购票的研究背景及现状，论述了本课题的研究意义，进行了系统需求分析，总体设计，详细功能设计与系统测试等。该系统在订票客户端实现了用户注册、用户登录、用户个人中心、电影详情浏览、影院场次选择、座位选择、支付购票、电影评价等功能。

在实现功能的过程中，通过自定义View优化了选座界面，利用MySQL数据库设计服务器端。本次开发的系统界面简单大方，操作便捷，高可用性。本文利用B/S设计模式，JAVA编程语言，对在线电影购票系统进行了研究。

关键字：影院购票，自助购买，JAVA，MySQL

Abstract

With the rapid development of China's film industry, the number of cinemas and the number of people watching movies are also on the rise, which leads to a series of problems, such as the increase of management cost of cinemas, the difficulty of users' buying tickets and the low efficiency of buying tickets. In order to reduce the cost of cinema management, improve the efficiency of management, and to facilitate the majority of users to buy movie tickets, the theater self-service ticket system came into being.

Firstly, this paper introduces the research background and current situation of online movie ticket purchase, discusses the research significance of this topic, and carries out system demand analysis, overall design, and detailed function design and system test. The system realizes the functions of user registration, user login, user personal center, movie details browsing, theater venue selection, seat selection, ticket payment, movie evaluation and so on.

In the process of realizing the function, the user-defined view optimizes the selection interface, and the MySQL database is used to design the server. The developed system interface is simple, easy to operate and highly available. In this paper, the B / s design mode and Java programming language are used to study the online movie ticketing system.

**Keywords:** Cinema ticket purchase, self-service purchase, Java, MySQL

目录

[1 绪论 1](#_Toc24573648)

[1.1项目背景和意义 1](#_Toc24573649)

[1.2国内外现状及分析 1](#_Toc24573650)

[2 系统相关技术介绍 4](#_Toc24573651)

[2.1 Java语言 4](#_Toc24573652)

[2.2 Eclipse开发平台 4](#_Toc24573653)

[2.3 JSP技术 5](#_Toc24573654)

[2.4 MySQL数据库 5](#_Toc24573655)

[3 系统分析 6](#_Toc24573656)

[3.1 可行性分析 6](#_Toc24573657)

[3.2系统功能需求 7](#_Toc24573658)

[3.3系统性能需求 8](#_Toc24573659)

[3.4 系统用例图 8](#_Toc24573660)

[4 总体设计 13](#_Toc24573661)

[4.1 系统设计原则 13](#_Toc24573662)

[4.2 系统功能模块图 13](#_Toc24573663)

[4.3 各模块基本流程图设计 14](#_Toc24573664)

[4.4 系统E - R图 16](#_Toc24573665)

[5 详细设计 17](#_Toc24573666)

[5.1动态建模 17](#_Toc24573667)

[5.1.1 用户登录模块 17](#_Toc24573668)

[5.1.2 用户模块动图建模 17](#_Toc24573669)

[5.2 数据库设计 19](#_Toc24573670)

[6 系统的实现 22](#_Toc24573671)

[6.1 系统主界面 22](#_Toc24573672)

[6.2 购票主界面 22](#_Toc24573673)

[6.3 确认购票信息主界面 23](#_Toc24573674)

[6.4 付款主界面 23](#_Toc24573675)

[6.5 用户注册界面 24](#_Toc24573676)

[6.6 用户登录界面 25](#_Toc24573677)

[6.7 用户个人信息界面 26](#_Toc24573678)

[6.8 用户修改密码界面 27](#_Toc24573679)

[6.9 用户购票界面 27](#_Toc24573680)

[6.10 管理员登陆界面 29](#_Toc24573681)

[6.11 管理员管理 30](#_Toc24573682)

[6.12 管理员电影票信息管理 31](#_Toc24573683)

[6.13 订单管理 31](#_Toc24573684)

[7 系统测试 33](#_Toc24573685)

[7.1 测试的主要内容 33](#_Toc24573686)

[7.2 测试与分析 33](#_Toc24573687)

[8结论 35](#_Toc24573688)

[参考文献 36](#_Toc24573689)

[致谢 37](#_Toc24573690)

# 1 绪论

随着我国社会经越来越多的人开始重视对精神文化的追求，电影正是精神文化里不可或缺的重要组成部分，周末节假日看电影已经被越来越多的人选择为假期休闲方式之一。因此，开发出适合大众需求的电影院自助购票系统是非常重要的。

## 1.1项目背景和意义

从我国1905年第一部电影《定军山》，到现在2019年，我国的电影产业已经有了一百多年的历史，从原先的露天放映机，到现在的各大影院；从原先的2D影片，到现在的3D影片；从原先的内容题材单一，到现在的内容题材包罗万象，人们有了越来越良好的观影体验，越来越多的人走进电影院观看自己喜欢或者感兴趣的电影，这就引起了购票难，排队等候时间长等问题的发生，而自助购票系统的诞生可以完美解决这些问题。

过去人们只能早早的到影院现场买票，这意味着人们即使提前去影院踩点排队，也并不能保证可以购到。如果碰到热门影片上映，电影院里更是场场爆满、一票难求。观众们可能提前了好多天去排队也买不到自己心仪的影票，这对于当下快节奏的生活来说，会造成大量且不必要的时间浪费，所以面向大众公布电影的排片情况和观影时的座位选择等就成了重中之重的问题。但是关于用户账号的问题，现在有一种方案，是以互联网终端特有的标识信息等等实现组合，这包括设备型号、IMSI码、IMEI码等，继而通过生成的终端身份信息来认证客户端用户的身份。这里的自定义生成身份信息的算法，可以提供给用户更好的体验方式，对安卓系统本身的安全性能来说也是提高了保障。

当电影购票与当代的互联网技术相碰撞的时候，网站端和移动端在线订购电影票系统就顺势出现了。

## 1.2国内外现状及分析

随着全球经济的不断进步，类似电影院这种中小型企业将会面临更加开放的市场环境和更加激烈的市场竞争。中小企业都已意识到信息化对企业发展的重要性。如何走适合自己企业发展的信息化的道路，使企业在激烈的竞争中立于不败之地，是摆在大都数中小民营企业的面前的新课题。

现在各影院的规模变得越来越大。如果还是按以前那种方法进行手工售票管理。一是工作效力比较低。另外也使得顾客等待时间过长，严重影响影院的利益。目前国内大多数的影院自动售票系统刚起步，远远不及欧美国家。本课题就是要将传统的影院人工售票程序化，用以提高工作效率。

不久的将来，传统的现场电影售票方式或将成为历史。本系统设计的缩略图选座功能，为大家在线购票、自由选择合适的观影位置提供了便利，美观而又简约的图形选座界面更是方便用户选择合适的观影位置，用户进入影厅时可以更快的找到自己订购的座位。用户在购票时需要输入有别于登录系统的支付密码，在用户的个人订票中心，客户端所有的交易记录都可以一一查看，让用户在使用本系统的过程中更加的安全与放心。该系统不仅让电影与人们的生活联系地倍加紧密，更给人们的生活带来了无限地便捷和愉悦的享受。

1.2.1 电影行业的国内外现状

2018年全球总票房340亿美元，中国内地电影的总票房就占了66亿美元，当年上映影片的总场次为7510万次，观影总人次达13.72亿，人均年观影一次。到了2018年，电影票房持续走高，国内票房更是高达79亿美元，同比2017年增长了20%，全球票房同比增长10%，刷新了多项票房纪录。从下表1-1中不难看出，越来越多的人选择去电影院观影。随之而来的就是各影院的售票问题，各大影院平台纷纷推出了各自的电影购票系统，以此来满足用户日益增长的需求。开发一个广大用户都适用的安卓在线电影购票系统，便成了各影院平台迫切需要解决的问题。

**表1.1 2017-2018年度国内外电影票房数据分析表（数据来源中国产业信息网）**

|  |  |  |  |
| --- | --- | --- | --- |
| 单位（亿美元） | 2017 | 2018 | 年度同比增长率 |
| 中国内地 | 66 | 79 | 20% |
| 海外 | 274 | 295 | 7% |
| 全球票房 | 340 | 374 | 10% |

1.2.2 电影票自助系统的研究现状

在计算机信息技术的发展如此迅猛的今天，早已可以足不出户就可以进行电影票的预定了。在这种大背景之下，基于JAVA的自助电影订票系统就应运而生了。

目前，应用市场上比较受欢迎的电影购票系统是由美团公司开发的猫眼电影，一般情况下只需要30-60秒就可完成选座购票流程，使用方便，而且价格更为便宜，深受用户们的喜爱。猫眼电影是一个集电影资讯、影票评分、电影衍生品销售、用户观影互动为一体的一个软件系统，同时，它可以给电影制片的发行方以及有相关合作的影院提供适合于各个消费人群的营销方案，助于各影片的票房热卖。另外，阿里旗下开发的淘票票也是广受人们的青睐，它是集影片评分和各种时下热门影片相关的资讯为一体的电影平台，这包括海量的影片预告片，主演们的高清海报与剧照、电影解读、电影评论评分等。

还有其他较受用户喜爱的购票软件有微信电影、拉手团购、时光网等，但是这些软件的用户圈还没有猫眼和淘票票广泛，因为这些软件都没有足够大的用户交互平台。用户在系统上在线购票后，无法在购票平台与其他用户们互相交流自己的观影体验和感受，用户关于电影和购票平台的建议也没有方式上传服务端。因此，开发一个便于用户间交流的JAVA电影购票系统具有十分重要的意义。

# 2 系统相关技术介绍

本系统是由现阶段较为成熟的Java编程语言，运用了面向对象的思想来进行开发的，基于JAVA1.8、Eclipse集成开发环境、Tomcat9.0、MySQL数据库来进行开发的。本系统通过Tomcat，来开发程序。

## 2.1 Java语言

作为一种面向对象的计算机语言，Java将所有东西都当成自己的对象。对象可被认为是一种创新型的变量，它能够存储数据，并且可以操作自身数据。如今，Java技术被广泛使用于大型应用程序的开发设计，正因其“一旦写成处处可用”的特点，Java技术已成为全球最热门的开发语言之一。尤其是在Web的应用领域，Java体现着它良好的使用性和简便性。

Java有如下特征：

1）编写简单：在语法上，Java有很多语句类似于C++，例如常用的循环语句和控制语句。而单从语言简单性来看JAVA比C++要简单，JAVA简化了部分C++的方法或是直接弃用。例如C++中指针的概念。

2）面向对象：每个类中包含多个该类的对象，每个对象都有属于自己的存储空间。另外，Java还支持封装现有的对象，来产生新型对象。现实生活中，我们把机动车看成是一个类，那么小轿车可以看成这个类中的一个实例，即JAVA中对象的含义。每个对象都有自己的属性，可以根据接收到的请求进行不同操作。

3）JAVA对平台无要求：只要配置好Java运行的环境变量、搭建好Java运行平台，即可实现JAVA程序的运行。

4）JAVA支持多线运行。使用Java语言编写的程序在运行时，处理器在处理时可以在多个线程之间直接快速地切换，由此给人一种多线程同时运行的感觉。然而C++则不提供这种多线程运行机制。

5）JAVA程序是动态的。JAVA支持对方法进行重写和重载。JAVA程序由一系列的类组成，在类中可以编写自己所需的方法，也可以导入Java原有的方法；用户可以根据需求以重载或重写的方式来修改程序，以此来完成对代码的动态维护。

## 2.2 Eclipse开发平台

Eclipse是著名的跨平台的自由集成开发环境（IDE）。其本身只是一个框架平台，但是众多插件的支持使得Eclipse拥有其他功能相对固定的IDE软件很难具有的灵活性。许多软件开发商以Eclipse为框架开发自己的IDE，因此得到了众多组织鼎力支持。

## 2.3 JSP技术

JSP（Java Server Pages）技术是由Sun公司发布的用于开发动态Web应用的一项技术。它简单、使用便捷；支持跨平台运行。JSP技术在一定程度上简化了浏览器应用程序的设计，高效灵活地实现与服务器的交互。利用JSP技术开发的Web程序可以实现服务器、浏览器便捷交互。现今，JSP在Web开发领域越来越受欢迎，许多大型的网站都采用JSP进行设计开发。

使用JSP技术，Web页面开发人员可以使用HTML或者XML标识来设计前端页面，结合JSP小脚本和标识内容可以生成动态的界面元素。（例如可以通过编写JavaScript可以动态地创建一个表格等）。生成内容的逻辑被封装于标签中，前端人员对可以编辑页面布局排版，而不会改变内容的生成方式。 JSP页面还能嵌套小脚本，而这些脚本将在服务器端运行。

在服务器端，JSP引擎解释JSP标识和小脚本，运行相应脚本，完成相应的请求，执行预设的操作；最终将操作的结果返回到前端网页。因此，JSP技术在一定程度上保护了开发源代码，同时又不影响页面的正常使用，还可实现跨平台运行。

与ASP相比，JSP能适应更多平台，因为大部分平台都支持JAVA语言。另外JSP通过JDBC连接数据库，实现对数据库的管理（包含对数据的增删改查等操作）。Sun公司提供了JDBC-ODBC的桥连方式,因此JAVA可以访问ODBC驱动下的数据库；常见的有Oracle、MS SQL Server 和Access等数据库。

## 2.4 MySQL数据库

MySQL是一个关系型语言数据库管理系统，由瑞语言典MySQL AB 公司语言开发，目前属于 Oracle 旗下产品。MySQL 是最流语言行的关系型数据库语言管理系统之一，在Web方面MySQL是最佳的选择。它具有以下特点：

 代码开源，无需付费即可使用，开发成本低。

 体积小，运行速度快。

 以标准的SQL语言实现对数据的管理。

 提供多种应用程序接口，包含JAVA、C++、PHP等。

 拥有多种连接途径，包含JDBC、TCP/IP等

# 3 系统分析

系统需求分析是在系统开发前的一个重要的活动，同时也是软件生存周期中的一个重要环节。系统需求分析阶段需要完成的工作是在结合实际与需求的情况下，理论上总结出该系统具体需要实现什么功能，确认后整理成完整清晰的要求并记录下来，供以后真正开发系统时开发时参考。只要做好系统分析，就能做出符合自己需求的系统。

## 3.1 可行性分析

可行性分析是对系统开发是否可行的一种有效分析，主要分为技术可行性、经济可行性和操作可行性三点。

3.1.1 技术可行性

在做系统之前首先要清楚的知道这个系统做出来有什么功能，能做什么样的事，达到什么样的目的。还应该了解目前市场上类似的软件有什么特点和功能，才能使自己的系统有自身的特色。

本系统采用的架构模式是JAVA系统的分层模式：应用层、应用框架层、系统运行库层和Linux核心层，每一层都只专注于当层所需提供的服务即可，且各层服务器所需的服务皆自下而上。

从最底层的Linux内核层分析来看，可以说整个JAVA的操作系统都是基于Linux内核来建立的，它为操作系统的内存管理、内核驱动程序、进程管理、硬件交互、以及中断管理等提供了最基础且最核心的功能。

为系统的应用框架层提供支撑的就是JAVA的系统运行库层，应用框架层需要调用系统的组件，它所提供的API可以直接调用，以及访问应用程序的核心，我们大多是基于Application Framework进行开发和研究应用程序的。

一般都是在应用层来完成JAVA的应用开发，我们所开发的这个应用程序，即在线电影购票系统就是在最高层应用层来运行，系统里的编程语言都运用了Java来实现。当然，应用层也有JAVA系统自带的部分核心应用程序，但是可以说所有的应用程序，都是通过JAVA系统的应用程序编程的接口函数来实现的。

综合技术层面的分析而言，本购票系统是以JAVA 的开发环境来完成客户端的设计，并借用了JAVA系统的分层架构，用MySQL来开发订票系统的后台数据库，使用Java作为此次研发编程的系统语言。纵观整个系统的开发来说，从技术到语言都非常的贴近当前的高端开发水平，在网站终端成功实现用户界面与服务器之间的无缝交互，与良好的开发环境以及十分合适的开发结构是分不开的。所以不得不说，技术开发的可行性层面，也是实际可行的。

3.1.2 经济可行性

电影院需要配备相应的自助购票机来应用此系统，并且需要定期对机器进行维护保养，相比于人工售票来说节约了人力成本。

通过以上分析，本系统在经济性上是完全可行的。

3.1.3 操作可行性

用户可以使用本自助购票系统查询影片具体信息及评分和宣传海报，也可以注册成为用户享受影院购票的特有折扣，还可以查询自身账号的具体信息并进行购票充值。本自助购票系统操作界面清晰，操作步骤简单，各个年龄段的用户都可以独立使用，不存在操作不懂的问题。可以满足广大用户的购票需求，能够解决实际生活中的问题，并且也适用于影院购票系统的实际应用。通过以上分析，本系统在操作性上是完全可行的。

## 3.2系统功能需求

影院购票系统自助购票系统主要由显示实时时间、购票、确认购票信息并扫描二维码付款、用户登录、用户注册、用户信息管理、用户购票和用户确认信息并从账户余额扣款几个模块构成。

在此次开发的基于JAVA的电影购票系统中，依据功能需求的分析，划分了两大模块，即用户模块与电影模块两部分。

用户如果有订票的需求时，就必须先登录到个人的系统账号，因此用户只要想对系统进行更多的操作时，都必须先注册属于用户个人的购票账号。用户成功注册购票账号后，系统都会自动为用户提示成功信息；相应的，一旦注册失败，系统同样也会给出相应的提示，例如该用户名已被注册，或两次密码不同等，以此来方便用户们注册使用。当用户注册成功以后，就可以登录自己的账号来操作本系统了，包括电影场次的选择，地区影院的选择，浏览影片的详情，查看电影评分，收藏自己喜爱的电影，以及通过图形界面可视化的选择自己心仪的座位等[6]。用户选定座位以后可以选择不同的支付方式，包括支付宝、微信、或银行卡，验证支付密码后即可订票成功，这也是出于用户的支付方便以及账户安全的角度来考虑。此外，如果用户在没有订票的情况下进行评论，该条评论记录上会被标注为未订票。

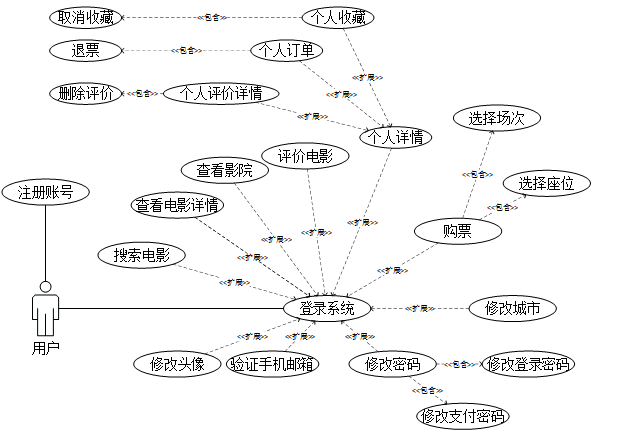
不可避免的是电影的上映场次和时间都有期限，一旦过期，影片就会自动从系统中下架删除。在用户角度来说，电影购票系统不仅可以方便的查看正在热映的影片信息，还可以查看未来上档的影片讯息，包括影片的超清海报，有助于用户知晓相关影片的内容详情，如电影的导演，参演人员等。同时，本系统还会提供订购过这部电影的人的观影评价，可供用户们在购票选择时做出简单的参考。

## 3.3系统性能需求

电影院自助购票系统对兼容性和安全性有着非常高的要求，所以整个系统应当性能良好、安全可靠、可以满足大量用户同时登录系统购票。此外系统应当操作流畅、界面清晰、维护简便。数据库要求运行速度快，稳定性、安全性耐用性高。

## 3.4 系统用例图

根据上文描述的系统功能需求，画出了基于JAVA的电影购票系统的用例图，图3-1是用户的用例图。



**图3-1用户用例图**

依据详细的功能需求，我们围绕了与用户相关的多个模块的用例图进行详尽的说明，可以参考如下表3.1，3.2，3.3，3.4，3.5，3.6，3.7，3.8，其分别为注册账号、修改城市（省市）、修改头像、修改登录密码、电影详情、收藏电影、用户选座、以及影片评价功能的详细说明。

|  |  |
| --- | --- |
| 用例类型 | 注册账号 |
| 功能简述 | 如果用户不注册账号，则只能简单查看电影的相关资讯，不可进行其他的系统类操作。 |
| 相关方 | 用户，计算机（服务器端），手机（客户端） |
| 前提 | 计算机服务器端正常，卡卢琳电影购票系统正常运行，用户了解该系统的注册流程。 |
| 主要流程 | 进入系统后，不管登录与否，用户都可以查看各种影讯资料。当用户有其他更多的使用需求时，就必须先注册独一无二的账号。这需要用户填写系统唯一没被占用的用户名，密码，手机号码等个人基本信息，即算注册完成。 |
| 其他可能 | 从未注册过本系统；不按要求键入账号、登录密码、手机号码等重要信息；系统闪退等自身错误，添加用户操作无法成功；注册数据提交失败，数据库的添加出现错乱。 |
| 结果 | 成功注册，数据更新到系统数据库，新添加用户资料被数据库录入。 |

**表3.1 注册账号的用例图叙述**

**表3.2 修改城市的用例图叙述**

|  |  |
| --- | --- |
| 用例类型 | 修改城市（省市） |
| 功能简述 | 登录账号后可以自由修改城市，以及选择对应城市的影院。 |
| 相关方 | 用户，计算机（服务器端），手机（客户端） |
| 前提 | 用户已经成功注册、登录，计算机服务器端正常，卡卢琳电影购票系统客户端正常，用户知晓修改省市的流程。 |
| 主要流程 | 注册并登录成功，用户可以行使更多权限操作购票系统，如修改所在城市，即用户随时自主的选择自己的目标城市。 |
| 其他可能 | 并未注册过个人账号；系统出现闪退等错误，客户端无法即时修改城市；提交的修改城市数据失败，更新出错。 |
| 结果 | 修改城市（省市）成功，数据库同步成功，系统显示修改后城市。 |

|  |  |
| --- | --- |
| 用例类型 | 修改头像 |
| 功能简述 | 用户可以随时更换客户端头像。 |
| 相关方 | 用户，计算机（服务器端），手机（客户端） |
| 前提 | 用户已经成功注册、登录，服务器端运行正常，客户端一切正常，用户了解更换系统头像的流程。 |
| 主要流程 | 登录后，可以通过选择手机内部图片作为客户端头像。 |
| 其他可能 | 未注册过系统账号；系统出现闪退等问题，更换头像失败；修改后数据无法提交，数据库添加时出错。 |
| 结果 | 用户头像更换成功，并更新至数据库。 |

**表3.3 修改头像的用例图叙述**

|  |  |
| --- | --- |
| 用例类型 | 修改登录密码 |
| 功能简述 | 用户可以自由修改软件的登录密码。 |
| 相关方 | 用户，计算机（服务器端），手机（客户端） |
| 前提 | 用户已注册、登录系统，服务器端正常，客户端正常，用户熟悉更改登录密码的流程。 |
| 主要流程 | 用户可以通过忘记密码来修改密码，只需填写自己的注册手机号和邮箱，系统自动与数据库匹配是否一致即可。 |
| 其他可能 | 从没有注册过账号；系统出现卡顿等错误，不能正常进行修改密码；修改数据无法提交，数据库更新总出错。 |
| 结果 | 用户登录密码修改提交成功，并更新至数据库中。 |

**表3.4 修改登录密码的用例图叙述**

**表3.5 电影详情的用例图叙述**

|  |  |
| --- | --- |
| 用例类型 | 电影详情 |
| 功能简述 | 用户对影片有较少的了解时，就可以通过影片资讯来找出喜欢的影片。 |
| 相关方 | 用户，计算机（服务器端），手机（客户端） |
| 前提 | 登录正常，服务器正常，客户端正常，用户知道查看资讯的方法。 |
| 主要流程 | 用户可能对影片无太多了解，可能需要查看影片的详细信息，包括电影的导演，参演人员，影片简介等相关资讯。更重要的是，系统支持查看其它用户的观影评分和观影感受发布，帮助用户参考。 |
| 其他可能 | 非用户登录；数据库发生读取错误，无法对应影片的资讯；影片海报读取错误，导致系统端图片无法显示；无法看到其他用户的评分和评价；用户不注册也可查看影片的具体资讯。 |
| 结果 | 查看影片详细的资讯成功。 |

|  |  |
| --- | --- |
| 用例类型 | 收藏电影 |
| 功能简述 | 用户只有登录后才可以把自己喜欢的影片收藏至收藏夹，便于快速找到那个收藏的影片。 |
| 相关方 | 用户，计算机（服务器端），手机（客户端） |
| 前提 | 注册登录，服务器端网络正常，客户端正常，用户知道如何收藏影片。 |
| 主要流程 | 用户登录之后，进入电影详情页面，可以收藏电影，方便下次找到电影选座购票，节省时间。 |
| 其他可能 | 并非系统用户；没有注册账号；系统卡顿或闪退，无法将影片收藏至收藏夹内；提交收藏数据一直失败，数据库添加失败。 |
| 结果 | 用户收藏电影成功，并更新到数据库中。 |

**表3.6 收藏电影的用例图叙述**

|  |  |
| --- | --- |
| 用例类型 | 用户选座 |
| 功能简述 | 用户可以通过图形化界面选择观影座位。 |
| 相关方 | 用户，计算机（服务器端），手机（客户端） |
| 前提 | 注册登录，服务器提供的网络正常，客户端正常运行，用户了解选座流程。 |
| 主要流程 | 用户只有登录之后，才可以进行影院和场次的挑选，继而可查看已选场次的剩余座位情况，并为自己选择观影座位，一次最多可选三个座位。 |
| 其他可能 | 非用户登录；未注册账号；系统出现闪退或卡顿，无法放大缩小选座界面；  座位数据提交失败，数据库更新频繁出错。 |
| 结果 | 选座成功，并将座位信息更新至数据库。 |

**表3.7 用户选座的用例图叙述**

**3.8 影片评价的用例图叙述**

|  |  |
| --- | --- |
| 用例类型 | 影片评价 |
| 功能简述 | 当用户观影结束后，可对电影内容进行星级评分和评价，将自己的观影感受与其他用户进行分享。 |
| 相关方 | 用户，计算机（服务器端），手机（客户端） |
| 前提 | 注册登录完成，服务器网络正常，客户端运行正常，用户熟悉评价流程。 |
| 主要流程 | 登录进入系统，可对电影做出评价和评分，直观的为其他用户选择影片时提供参考。 |
| 其他可能 | 非用户登录；未注册账号；系统出现卡顿现象，无法发布评价和影片评分；  影评的数据提交失败，更新数据出错。 |
| 结果 | 影片评价记录添加成功，并将新数据更新至数据库中。 |

# 总体设计

根据系统需求分析，可以得出系统各个模块的设计思路，本章通过流程图的方式展现了用户管理模块、购票模块和影片信息模块的设计理念。

## 4.1 系统设计原则

本自助购票系统的设计原则主要遵循实用性原则、用户界面设计原则、可扩展性和可维护性原则和数据库设计原则。

（一）实用性原则

应该最大限度的满足实际工作的需要，充分考虑影院的实际需求，要把实用性原则发挥到最大化，将实用性放在第一要素考虑。

（二）用户界面设计原则

作为本自助购票的使用者，用户的体验是至关重要的。用户的目的是什么？是为了购买电影票，所以应该将系统界面做的简明清晰，能让用户直观的满足自身需求。操作界面应当简洁、明了、大方，符合大众的审美要求，让用户完成一次完美的购票体验是最终的目的。

（三）可扩展性和可维护性原则

为了适应将来的发展，本自助购票系统应当具有良好的可扩展性和可维护性。本自助购票系统的可扩展性应当包括能接纳现如今已有的系统和完美适配于今后新开发的系统。

（四）数据库设计原则

数据库作为购票系统的数据储存地应当体现安全性、完整性、一致性、规范性和可伸缩性。因为购票人数众多，所以要能够符合用户基数大这个要求，避免数据的丢失遗漏。

## 4.2 系统功能模块图

影院购票系统自助购票系统主要功能模块包含购票模块和用户购票模块，功能模块结构示意图如图4-1所示。



**图4-1 系统功能模块图**

## 4.3 各模块基本流程图设计

本自助购票系统主要分为用户管理模块、购票模块和影片信息模块。

.4.3.1 用户管理模块

用户管理模块由账户注册、修改密码和充值模块构成，如图4-2所示。



**图4-2 用户管理模块图**

4.3.2 购票模块

购票模块由选择影片信息、选择播放影厅、选定观影座位和付款等模块构成，如图4-3所示。



**图4-3 购票模块图**

4.3.3 影片信息模块

影片信息模块由影片海报、影片名称、影片导演、影片主演、影片产地、影片类型、影片时长和影片评分构成，如图4-4所示。



**图4-4 影片信息模块图**

## 4.4 系统E - R图

E-R图也称实体-联系图(Entity Relationship Diagram)，是描述现实世界关系概念模型的有效方法。是表示概念关系模型的一种方式。本自助购票的实体类型有、用户、电影票、密码和账户余额。可以购买电影票或者注册成为用户，用户可以购票电影票、修改密码和进行充值，如图4-5所示。



**图4-5 系统E-R图**

# 详细设计

## 5.1动态建模

### 5.1.1 用户登录模块

用户可以使用的身份去浏览影片信息和影院的排片信息，但当订购电影票的时候，登录账号，系统将会账号密码信息与数据库中的数据是否匹配。如果身份信息通过验证，用户就可以进入系统并可执行购票退票等一系列操作；相反，数据库比对失败，系统会给出登录不成功的原因，以此来帮助用户成功登录到系统内部。如图5-1所示。



**图5-1用户登录活动图**

用户登录系统需要经过如下：

1）登录首页，填入账户密码；

2）自动检测账户的身份信息是否与数据库匹配；

3）如若身份信被息验证成功，则会进入用户各自的页面；如若身份信息的验证不成功，则会弹出提示框，告知用户验证失败的原因，来帮助登录。

可能存在如下情况：

1）状态，即只是点开系统首页，未操作；

2）登录进系统，即根据账户和密码，通过数据库比对来验证身份是否正确；

3）无法登录，即用户输入了正确的账户和密码，由于一系列原因没有通过系统的身份验证。

用户登录状态图，如图5-2所示。



**图5-2 用户登录状态图**

### 5.1.2 用户模块动图建模

本系统的用户在登录成功时，进入到系统主页面，这时候可以对订购的影票信息做出选择了，点击确认购票，最多可订购3张电影票等，如果操作符合系统规范，则完成购票；相反，如果不符合系统流程，也会附上相应的提示，帮助你顺利订票成功。用户购票活动图，如图5-3所示。



**图5-3用户购票活动图**

在用户购票模块中，主要包括如下步骤：

1）已登录，进入电影选择界面；

2）选择喜欢的电影名、时间、座位等信息；

3）如果系统验证身份信息成功，数据库中将会自动插入本次的购票信息，并跳出弹框提示订票成功。用户购票序列图，如图5-4所示。



**图5-4用户购票序列图**

在系统的用户模块中，用户需先订购当前场次的电影的票，才可发表评论，用户评论电影主要包括如下几种情况：

添加评论，选择星级，输入评价内容，即可；

完成评论，即评分与评论皆按照要求来，数据也正确，系统会给成功提示；

3）无法评论，即给分评论频繁出错，包括操作和数据，从而导致系统添加用户评论到数据库时失败。用户评价状态图，如图5-5所示。



**图5-5用户评价状态图**

## 5.2 数据库设计

该系统的后台数据库设计是以JAVADB为基础的，其中相关的数据库表格包括用户基本信息表、电影资讯表、影院表、影院排片表、影座表、收藏表、影评表、订单表、城市（省市）表等。

根据数据库结构，以及系统功能需求等，设计了本系统的数据库，分别如下表5.1到5.9所示。

**表5.1用户信息表**

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 允许NULL值 | 说明 |
| userid | Int | NO | 序号为系统账号，主键 |
| username | string | NO | 用户注册名 |
| image | string | NO | 用户头像 |
| paypassword | string | NO | 用户注册时，自己键入的支付密码 |
| loginpassword | string | NO | 用户注册时，自己键入的登录密码 |
| Email | string | NO | 用户注册时，自己输入的邮箱 |
| Phone | string | NO | 用户注册时，自己键入的手机号 |

**表 5.2 电影信息表**

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 允许NULL值 | 说明 |
| id | Int | NO | 为电影的编号，为主键 |
| Moviename | string | NO | 电影的名称 |
| image | string | NO | 电影的相关海报图 |
| starttime | timestamp | NO | 电影的放映时间 |
| rate | float | NO | 电影评分 |
| ratenum | Int | NO | 电影评价的人数 |
| director | string | NO | 导演 |
| actors | string | NO | 主演 |
| collectednum | Int | NO | 影片收藏的数量 |

**表5.3影院信息表**

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 允许NULL值 | 说明 |
| id | Int | NO | 为影院的编号，为主键 |
| Cinemaname | string | NO | 影院的名称 |
| City | string | NO | 影院所在的城市 |
| Location | string | NO | 影院的具体位置信息 |

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 允许NULL值 | 说明 |
| id | Int | NO | 为排片的编号，为主键 |
| Movieid | Int | NO | 电影的ID编号 |
| Cinemaid | Int | NO | 影院的ID编号 |
| Hall | string | NO | 影院的厅名 |
| time | timestamp | NO | 影片的放映时间 |
| cost | Float | NO | 影片的单价 |

**表5.4影院排片信息表**

**表5.5影院座位信息表**

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 允许NULL值 | 说明 |
| id | Int | NO | 为座位信息的编号，为主键 |
| Cinemaid | Int | NO | 影院的ID编号 |
| Hall | string | NO | 影院的厅名 |
| time | timestamp | NO | 影片的放映时间 |
| Seatrow | Int | NO | 座位行号 |
| Seatcol | Int | NO | 座位列号 |
| ordered | bool | NO | 座位是否已售 |

**表5.6电影收藏信息表**

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 允许NULL值 | 说明 |
| id | Int | NO | 为收藏信息的编号，为主键 |
| Movieid | Int | NO | 电影的ID编号 |
| Userid | Int | NO | 用户的ID编号 |

**表5.7电影评价信息表**

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 允许NULL值 | 说明 |
| id | Int | NO | 为评价信息的编号，为主键 |
| Movieid | Int | NO | 电影的ID编号 |
| Userid | Int | NO | 用户的ID编号 |
| Evaluation | String | NO | 影片的评价内容 |
| Rate | Float | NO | 影片的评分 |
| Time | timestamp | NO | 评价时间 |

**表5.8订单信息表**

|  |  |  |  |
| --- | --- | --- | --- |
| **列名** | **数据类型** | **允许NULL值** | **说明** |
| id | Int | NO | 为订单信息的编号，为主键 |
| Userid | Int | NO | 用户ID编号 |
| Movieid | Int | NO | 电影ID编号 |
| Cinemaid | Int | NO | 影院的ID编号 |
| Hall | string | NO | 影院的厅名 |
| showtime | timestamp | NO | 影片的放映时间 |
| time | timestamp | NO | 购票的时间 |
| Seatrow | Int | NO | 座位行号 |
| Seatcol | Int | NO | 座位列号 |
| Cost | float | NO | 影片的单价 |

**表5.9城市信息表（省市）**

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 允许NULL值 | 说明 |
| id | Int | NO | 为省市对应的编号，为主键 |
| Province | string | NO | 省名 |
| City | string | NO | 市名 |

# 系统的实现

本章主要介绍了用户登录、用户注册、用户信息管理、用户购票和用户确认信息并从账户余额扣款等各个模块功能的具体实现，通过界面的截图和主要代码的展示来详细介绍。

## 6.1 系统主界面

在系统的主界面，用户可以根据自身需要选择要进行的操作，如果是普通用户，可以直接点击“普通用户购买”按钮进入购票界面，也可以点击“注册账户”按钮进入注册界面，用户注册的用户名唯一，当已被其他用户使用时，系统会自动为用户弹出提示框“用户名已被使用！”，。注册时需要输入两次密码，一旦两次密码不一致，就会告诉我们“确认密码和输入密码不一样！”，且两次密码输入皆不可见，确保了密码输入时不会泄露。如图6-1所示。



**图6-1 系统主界面**

## 6.2 购票主界面

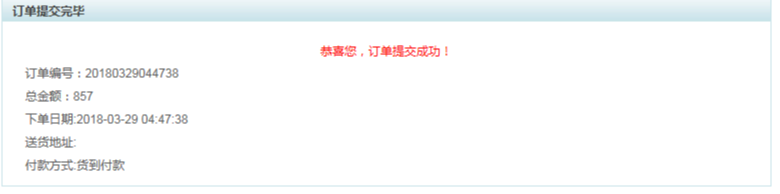
系统的购票界面，。中间部分是今日的影片列表、影片详情、播放影厅和座位表，如图6-2所示。



**图6-2 购票主界面**

## 6.3 确认购票信息主界面

当用户确定选择完成后，可以点击下方的“确定”按钮，进入确认购票信息主界面，如果用户确认有误，可以点击下方“返回”按钮，返回到购票主界面重新选择购票信息，如果用户确认无误，可以点击下方“我已确认”按钮，进入付款界面，如图6-3所示。



**图6-3 确认购票信息界面**

## 6.4 付款主界面

本自助购票系统采用当下最流行也是最普遍的扫描二维码方式进行付款，用户可以扫描二维码完成付款，付款完成后点击退出，系统会返回主界面，如图6-4所示。



**图6-4 付款界面**

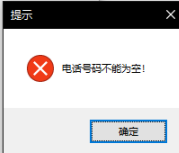
## 6.5 用户注册界面

在这个界面，用户可以输入对应信息，如图6-5所示。



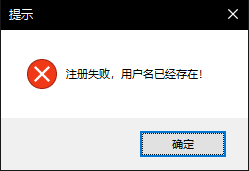
**图6-5用户注册界面**

如果有哪一项未输入，则系统会弹出提示，以未填联系电话为例，如图6-6所示。



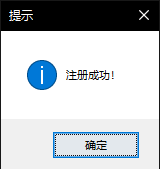
**图6-6漏填信息提示界面**

所有信息都填完之后，点击“确认”按钮，如果用户名重复的话则会提示“注册失败，用户名已经存在！”如图6-7所示。



**图6-7 用户名重复提示**

如果用户名没有重复的话则会提示“注册成功！”如图6-8所示。



**图6-8 注册成功提示**

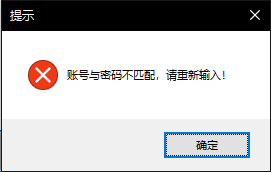
## 6.6 用户登录界面

用户可以点击“登陆”按钮来进行登录操作，系统会进入登录界面，用户需要输入信息进行登录，如果用户名和密码中任一项没有填写，登录按钮则呈现无法被点击状态。如图6-9所示。



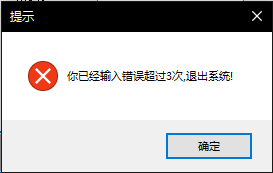
**图6-9 用户登录界面**

当用户输入完成后，点击登录按钮，如果用户名密码不匹配，系统会清空已输入的用户名和密码并提示用户输入错误次数，如图6-10所示。



**图6-10 账号密码不匹配**

如果输入错误次数超过3次，则系统会自动退出至主界面，如图6-11所示。



**图6-11 错误次数超过3次，退出系统**

主要代码如下：

//判断账号密码是否匹配

myCom.CommandText = "Select \* from InfoTable where username='" + textName.Text + "'and password='" + this.textPassword.Text + "' ";

## 6.7 用户个人信息界面

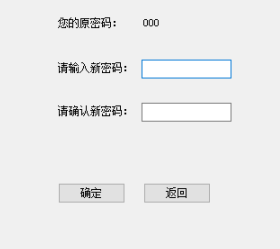
用户登录成功后，系统会自动进入用户的个人信息界面。在这里，用户可以进行购票，充值，修改密码操作，如图6-12所示。



**图6-12 用户个人信息界面**

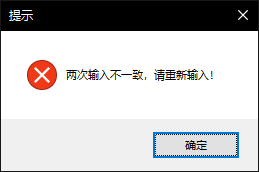
## 6.8 用户修改密码界面

用户点击“修改密码”按钮后，系统会进入修改密码界面，用户可以输入新密码并确认。当用户输入完成后，点击“确定”按钮，系统会提示修改成功并进入用户登录界面，再次让用户重新登录的目的是为了让用户更加熟悉自己的新密码。如图6-13所示。



**图6-13 用户修改密码界面**

如果两次输入不一致，系统会提示用户两次输入的新密码不一致，如图6-14所示。



**图6-14 输入不一致提示**

主要代码如下：

//修改密码操作

myCom.CommandText = "update InfoTable set password= '" + ensureNewpw.Text.Trim() + "' where username='" + value2 + "'";

## 6.9 用户购票界面

用户在选中要看电影后，选择数量即可，如图6-15所示。



**图6-15 用户购票界面**

待用户选择好购票信息后，点击“预定”按钮，系统会进订票车信息界面，如图6-16、6-17所示。



**图6-16 用户确认购票信息界面**



**图6-17 订单确认购票信息界面**

在用户付款之后，就可以在订单中心中看到付款成功的界面。如图6-18所示。



**图6-18 订单中心**

用户如果还想购买其他场次的电影票，就可以在旁边的搜索框里面搜索要看的电影，然后进行上述的步骤进行购买。

到此用户的功能基本实现，就可以看到用户的订单信息。只需要管理员在后台进行相关的确认即可。如图6-19所示。



**图6-19 订单确认**

## 6.10 管理员登陆界面

管理员可以根据自己的信心登陆电影院自助购票系统进行系统的管理，登陆界面如图6-20所示。



**图6-20 管理员登陆**

## 6.11 管理员管理

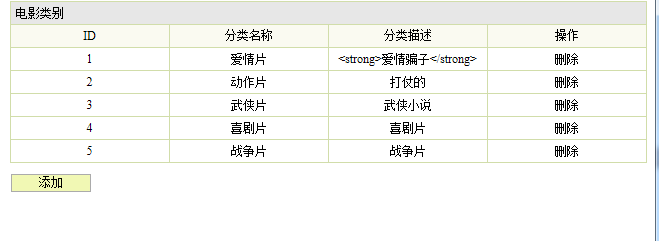
同样，只有在管理管的用户和密码信息和数据库中的信息想对应，才可以登陆成功，如果输入错误可以点击重置键，对文本框内的信息重置。

登陆成功后，管理员可以查看系统的相关信息，如图6-21所示。



**图6-21 系统信息查看**

管理员可以在电影类别中对现有的电影类别进行相应的信息修改和添加删除操作。如图6-22所示。



**图6-22 电影类别**

还可以添加相关类别的描述等信息，如图6-23所示。



**图6-23 类别信息**

## 6.12 管理员电影票信息管理

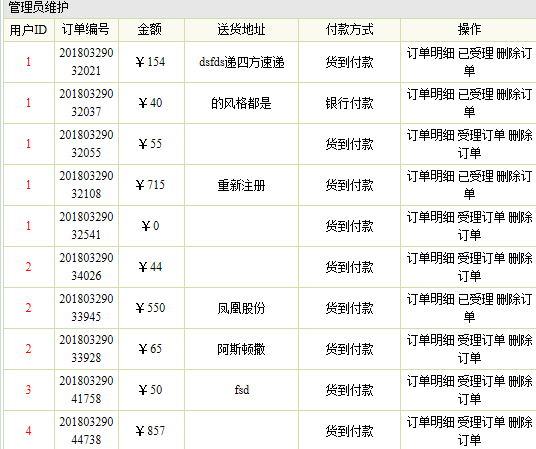
管理员可以在电影票管理中对相应的电影票进行相关的信息设置，例如这只电影票的价格和添加电影和删除电影等操作。如图6-24所示。



**图6-24 电影票信息管理**

## 6.13 订单管理

管理员要在订单管理界面对用户的相关操作进行管理，例如电影票的发放，本系统暂时完成的是送票，但是实现现场取票是必须实现的功能。另外，用户的付款方式和付款情况在这个界面也会详细地呈现给管理员，管理员还可以看订单地详情，以及对不合理订单地删除等操作。如图6-25所示。



**图6-25 订单管理**

# 7 系统测试

在自助购票系统设计开发完成后，需要对其进行测试分析，目的在于检验系统的可实施性以及各方面还有可能存在的问题。系统测试由测试的主要内容和分析构成。

## 7.1 测试的主要内容

测试的主要内容主要有用户登录测试、用户购票测试和用户注册测试，如表7.1所示。

**表7.1 测试用例**

|  |  |  |  |
| --- | --- | --- | --- |
| 测试用例 | 步骤 | 预期结果 | 实际结果 |
| 用户登录测试，如果用户名和密码都为空。 | 不输入用户和密码 | 登录按钮为不可点击状态 | 与预期相符 |
| 用户购票测试，如果不选择影片和播放影厅。 | 不选择影片和播放影厅 | 确定按钮为不可点击状态 | 与预期相符 |
| 用户登录测试，如果用户名是正确的，密码是错误的。 | 输入正确的用户名和错误的密码，点击登录按钮 | 登录失败并提示 | 与预期相符 |
| 用户注册测试，如果未输入其中某一项 | 输入除联系电话以外所有项，点击提交注册按钮 | 登录失败并提示 | 与预期相符 |
| 用户注册和登录时未输入验证码或输入验证码错误测试 | 输入错的验证码或不输入，点击登录或提交注册按钮 | 登录和注册失败并提示 | 与预期相符 |

## 7.2 测试与分析

经过测试，本自助购票系统运行情况良好，占用系统资源较少，界面清晰，操作简单便捷。但是，本自助购票系统也存在一些不足之处：

1、本自助购票系统采用SQL Server 2014作为数据库，所存储的容量有限。

2、本自助购票系统的功能还不够强大，不能够满足于有较高要求用户的需求；

3、本自助购票系统的操作界面还有较大缺陷。

# 8结论

本自助购票系统的前端使用了JAVA作为开发语言，后台数据库使用MySQL来储存数据。

本自助购票系统的实现操作庞杂，很多功能都是互相有联系的，如用户的注册和登录等。对电影院购票流程不熟悉是本自助购票系统的欠缺。各个模块之间缺乏必要的联系，整个系统显得生硬。在开发设计过程中，由于开发经验不足遇到了许多问题，最终还是在导师的细心指导下，通过自己的努力，使问题得到了解决。由于开发经验的欠缺，本系统还存在很多不足之处。其次，本系统还只是一个比较局限的系统，只能完成比较有限的功能，希望通过对国内外领先的相关技术的学习和以后工作中得到的锻炼可以将现在系统统进一步提升，使之成为功能更加全面、使用更加合理、可用性更高的电影院自助购票系统。

因为实习的问题，导致我开始做毕设的时间比较晚，外加上我自身的技术不太好，基于JAVA的电影购票系统有些预期的功能并未实现，例如支付功能未能与银行对接，且系统的界面设计较为简单、系统数据的安全问题还未解决、客服功能未实现等等。所以在后面的学习中，我会不断地加强自己的专业知识。虽然本次开发的系统在众多影购票系统之中只是冰山一角，但是伴随互联网的快速发展，以及现在选择进入影院观影的人越来越多，电影购票系统今后发展的空间还是巨大的，为了以后让更多人使用在线影购票系统，今后还要不断地进行完善和改进。

# 参考文献

1. 孟庆昌．ASP.NET网站开发先锋． 北京：机械工业出版社，2010
2. 韩颍，卫琳，谢琦．ASP.NET 4.5动态网站开发基础教程．北京：清华大学出版社，2015
3. 张昌龙，辛永平．ASP.NET 4.0从入门到精通．北京：机械工业出版社，2011
4. 张跃廷，王小科，许文武．ASP.NET数据库系统开发案例精选．北京：人民邮电出版社，2007
5. 宋海兰，李航，沙继东．ASP.NET 3.5项目开发实战（第二版）．北京：电子工业出版社，2013
6. 王喜平，于国槐，宋晶．ASP.NET程序开发范例宝典．北京：人民邮电出版社，2015
7. 徐贤亮．合肥众赢公司员工信息管理系统的设计与实现．合肥：大连理工大学，2013
8. 石楚韵．北京轨道交通公司员工管理信息系统的设计与实现．成都：电子科技大学，2014
9. 炎士涛，陈文臣，胡荷芬，张彬．ASP.NET项目开发案例精粹．北京：电子工业出版社，2011
10. 高宏，李俊民．ASP.NET典型模块与项目实战大全．北京：清华大学出版社，2012
11. 丁士锋，蔡平．ASP.NET项目开发指南．北京：清华大学出版社，2010
12. 赵燕．企业人事信息管理系统设计与实现．成都：电子科技大学，2010
13. 邓文渊，陈惠贞，陈俊荣．ASP.NET与网络数据库技术．北京：中国铁道出版社，2010
14. 丁士锋．亮剑ASP.NET项目开发案例导航．北京：电子工业出版社，2012

# 致谢

感谢恩师对于我毕业设计的细心指导与建议。因为老师的鼓励，我克服了许多困难，顺利成功完成了系统。我所学到的不仅是知识，而且是对生活的态度，一种积极的、进步的、一丝不苟的生活态度。感谢老师您对本项目的大力支持。在这个项目中遇到的许多问题解决过程中，同学们的帮助对我帮助很大。

感谢答辩会所有的老师对于我的论文提出的宝贵意见，让我能够充实自己。

感谢我的父母在大学和项目期间对我的关心，同学在这几年中给了我帮助。

在这篇论文的开发过程中，我得到了导师的技术指导，我要特别感谢我的导师对我的大力帮助。指导老师非常的耐心和大力的支持我，在所有的系统开发框架的总体结构和功能设计上给了我很大的帮助，同时，给了我们这么好这么安静的环境学习，在规划设计过程中对我进行的耐心教导并讲解工作的许多细节，现在我可以完成我的毕业论文这些因素都起到了决定性的作用。