# 基于js的电影售票系统 （初稿）

说明：本文档参考国标“13 - 软件(结构)设计说明(SDD)”和案例中“四、网上购书系统概要设计说明书”等资料自定而成。

目录

[基于js的电影售票系统 （初稿） 1](#_Toc71400398)

[1. 引言 3](#_Toc71400399)

[1.1标识 3](#_Toc71400400)

[1.2系统概述 3](#_Toc71400401)

[1.3文档概述 3](#_Toc71400402)

[1.4基线 3](#_Toc71400403)

[2.引用文件 3](#_Toc71400404)

[3.可行性分析的前提 4](#_Toc71400405)

[3.1项目的要求 4](#_Toc71400406)

[3.2项目的目标 4](#_Toc71400407)

[3.3运行环境 5](#_Toc71400408)

[3.4约束条件 5](#_Toc71400409)

[4.总体设计 5](#_Toc71400410)

[4.1基本功能 5](#_Toc71400411)

[4.2附加功能 5](#_Toc71400412)

[5. CSCI体系结构设计 6](#_Toc71400413)

[5.1体系结构 6](#_Toc71400414)

[5.2全局数据结构说明 6](#_Toc71400415)

[5.3CSCI部件 6](#_Toc71400416)

[5.3执行概念 7](#_Toc71400417)

[6. 后台数据表 7](#_Toc71400418)

[7. 尚未解决的问题 7](#_Toc71400419)

[8.注解 7](#_Toc71400420)

[9、附录 8](#_Toc71400421)

## 引言

### 1.1标识

系统标识符：

名称：基于Html的电影售票系统

畅影影业

版本号：1.0.0

保密级别：普通

### 1.2系统概述

应课程要求，建立软件工程初步项目。

本团队实现基于web的电影购票网站。

提出方要求：(1)支持用户在线浏览电影售票页面、热映电影信息、电影排行榜；

(2)支持用户搜索附近影院；

(3)支持用户在线购票、实时支付、退款请求等；

(4)用户在观影结束后，可以对影院环境以及影片质量进行评价。

(5)系统UI界面简洁大方，操作简单，实时更新电影影院数据；

项目目的：建立一个成规模的在线网络电影售票网址，实现上述功能。

实现环境为html环境，即完全架构在服务器上的浏览器运行模式。

限制条件大概限制为电脑用户，手机上运行此网页效果不太良好。

### 1.3文档概述

本文档适用于html电影售票：畅影影业（网站）。

此项目为学生课程设计项目，目前处于开发阶段。

投资方：暂无，可以规划为学生自身；

需求方：目标项目组成员以及老师助教验收用；

用户：学生（开放性网址，均可以登录并进行合法操作）；

开发方：本组组员；

支持机构：暂无

### 1.4基线

畅影影业可行性分析研究报告（FAR）

## 2.引用文件

《软件工程（第4版）》

《Architecture description template for use with ISO/IEC/IEEE 42010:2011》

《软件(结构)设计说明》(SDD, Software Design Document)

GB-T-8567-2006计算机软件文档编制规范

山东大学软件工程课程课件

## 3.可行性分析的前提

### 3.1项目的要求

实现以下全部功能。

1. 注册登入功能：

在登入页面（注册页面）登入账号（注册账号）即可。有管理员和普通用户两种登入方式，管理员登入可以对电影排版，哪个电影可以上映，在哪里上映等做出修改。

1. 买票功能：

登入之后，在首页的正在热映页面可以看到如今正在热卖的电影，可能会快速找到自己喜爱的电影。

我们也可以点击首页的右方电影按钮，然后在里面可以查看如今所有在映的电影。我们也可以在影院页面查看是否有自己喜欢的电影，每个影院都有自己上映的电影的清单。

最后我们可以直接在首页上方的搜索界面搜索想要看的电影。

在上述所有操作中只要找到想要购买的电影票，都可以在当前页面购买。

1. 搜索功能：

在首页上方直接搜索，可以搜索自己想看的电影，或者自己心仪的电影院。

1. 预购功能：

在首页的即将上映页面又即将上映的电影，我们可以在里面进行预购。

1. 查看订单功能：

在右下方“我的”界面可有“我的订单”功能，在这里我们可以看到自己的所有历史订单，清楚自己的每一笔消费。

1. 想看的电影：

也是在右下方“我的”界面右边“想看的电影”可以在那里写入想看的电影。

1. 看过的电影：

同是“我的”界面，我们可以查看自己所有看过的电影，留下自己的点点滴滴。

1. 评论功能：

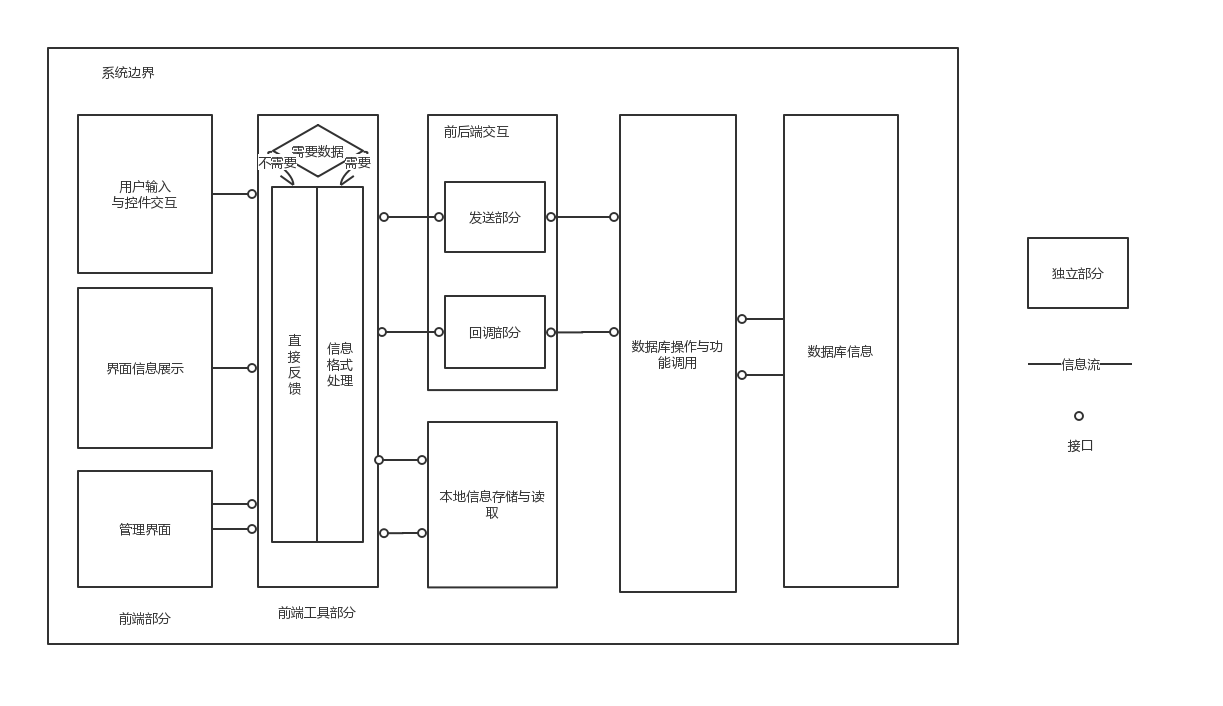
可以再电影下进行评论，谈谈自己对电影的感受等。

### 3.2项目的目标

a.本次开发的目标是完成开发一个能在很多平台上运行的电影购票网站，是为解决多种平台下的用户购票和选票的问题。网站可以为多种平台下的用户提供电影信息和评价等信息，方便用户选择电影。同时对于电影的放映时间和地点会进行同步更新。能够使用户更加全方面地安排自己地购票安排

b.购票网站要求实现的功能有搜索电影，查看电影信息、对电影进行评分、对电影进行评价、购票、选座、点赞、收藏等功能

c.本次开发产品为独立产品，系统的层次结构图如下



### 3.3运行环境

此次开发是以B/S架构为模型构建，硬件环境包括前端程序运行环境为客户端的浏览器、后端程序的运行环境为服务器。程序语言支持环境包括前端的HTML、CSS、JavaScript环境，以及后端的express运行环境。储存支持环境需要在后端配置Mysql环境。

### 3.4约束条件

本次开发的约束条件如下：经费限制不考虑、开发期限为10~12周、采用的方法和技术比较容易实现，不能投入过多的人员和时间。过滤掉政治导向不正确的曲目和言论等信息、遵循严禁盗用他人版权等法律限制、转播积极向上的正能量信息。

## 4.总体设计

### 4.1基本功能

(a) 基于web的电影购票网站

(b) 注册登录用户可在线浏览热映电影信息，电影排行榜，;

(c) 注册登录用户查看附近影院，支持在线购票，实时支付等；

(d) 用户在观影结束后，可以对影院环境以及影片质量进行评价

(e) 系统UI界面简洁大方，操作简单，实时更新电影影院数据。

(f) 具有良好的平台可移植性，支持多人同时在线访问。

(g) 网页刷新时间等待时间不超过2s；用户能够流畅使用各项功能。

### 4.2附加功能

1. 安全与保密功能系统涉及用户私密信息，对于整个系统，需要完整的权限控制，每个用户分配特定的用户视图所应有的访问权限，防止某些人恶意的攻击系统，修改原始记录。设置必要的异常处理机制，比如当出现系统未知的操作时。也会弹出相应的错误提示等，以尽可能地保证用户体验。
2. 客户账号管理功能：管理人员能够列表显示用户的帐号信息、禁用或删除选定的用户帐号，在收到客户请求之后也可更改客户的账号信息。
3. 通知功能：管理人员可通过软件内消息向用户发送通知。
4. 电子邮件/短信服务：管理人员能够在客户账号信息出现异常时通过绑定的电子邮箱/手机号码向用户发送邮件/短信提醒。
5. 评论维护功能：对于违规评论，用户可向管理人员发送反馈，反馈人数达到一定程度即向管理人员显示该条消息，管理人员可以对其进行删除并发送消息通知该评论所属用户

## 5. CSCI体系结构设计

### 5.1体系结构

#### 5.1.1程序(模块)划分

#### 5.1.2层次关系：

### 5.2全局数据结构说明

### 5.3CSCI部件

本条应：

a.标识构成该CSCI的所有软件配置项。应赋予每个软件配置项一个项目唯一标识符。

注:软件配置项是CSCI设计中的一个元素，如CSCI的一个主要的分支、该分支的一个组成部分、一个类、对象、模块、函数、例程或数据库.软件配置项可以出现在一个层次结构的不同层次上，并且可以由其他软件配置项组成.设计中的软件配置项与实现它们的代码和数据实体(例程、过程、数据库、数据文件等)或包含这些实体的计算机文件之间，可以有也可以没有一对一的关系。一个数据库可以被处理为一个CSCI，也可被处理为一个软件配置项。SDD可以通过与所采用的设计方法学一致的名字来引用软件配置项。

b.给出软件配置项的静态关系(如“组成”)。根据所选择的软件设计方法学可以给出多种关系(例如，采用面向对象的设计方法时，本条既可以给出类和对象结构，也可以给出CSCI的模块和过程结构)。

c.陈述每个软件配置项的用途，并标识分配给它的CSCI需求与CSCI级设计决策(需求的分配也可在6.a中提供)。

d.标识每个软件配置项的开发状态/类型(如新开发的软件配置项、重用已有设计或软件的软件配置项、再工程的已有设计或软件、为重用而开发的软件等)。对于已有设计或软件，本说明应提供标识信息，如名称、版本、文档引用、库等。

e.描述CSCI(若适用，每个软件配置项)计划使用的计算机硬件资源(例如处理器能力、内存容量、输入/输出设备能力、辅存容量和通信/网络设备能力)。这些描述应覆盖该CSCI的资源使用需求中提及的、影响该cscl的系统级资源分配中提及的、以及在软件开发计划的资源使用度量计划中提及的所有计算机硬件资源。如果一给定的计算机硬件资源的所有使用数据出现在同一个地方，如在一个SDD中，则本条可以引用它。针对每一计算机硬件资源应包括如下信息：

1)得到满足的CSCI需求或系统级资源分配；

2)使用数据所基于的假设和条件(例如，典型用法、最坏情况用法、特定事件的假设)；

3)影响使用的特殊考虑(例如虚存的使用、覆盖的使用、多处理器的使用或操作系统开销、库软件或其他的实现开销的影响)；

4)所使用的度量单位(例如处理器能力百分比、每秒周期、内存字节数、每秒千字节)；

5)进行评估或度量的级别(例如软件配置项,CSCI或可执行程序)。

f.指出实现每个软件配置项的软件放置在哪个程序库中。

A．标识符

B．静态关系：

C．用途

D．开发状态

E．使用资源

F．程序库放置

### 5.3执行概念

本条应描述软件配置项间的执行概念。为表示软件配置项之间的动态关系，即CSCI运行期间它们如何交互的，本条应包含图示和说明，(若适用)包括执行控制流、数据流、动态控制序列、状态转换图、时序图、配置项之间的优先关系、中断处理、时间/序列关系、异常处理、并发执行、动态分配与去分配、对象/进程/任务的动态创建与删除和其他的动态行为。

状态转换图：

## 6. 后台数据表

-- MySQL dump 10.13 Distrib 8.0.23, for Win64 (x86\_64)

--

-- Host: localhost Database: db\_film

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

-- Server version 8.0.23

--

-- Table structure for table `t\_admin`

--

DROP TABLE IF EXISTS `t\_admin`;

CREATE TABLE `t\_admin` (

`admin\_id` int unsigned NOT NULL AUTO\_INCREMENT COMMENT '管理员id',

`name` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '管理员姓名',

`password` varchar(100) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '管理员密码',

`sex` varchar(10) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '管理员性别',

`phone` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '管理员联系电话',

`birthday` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '出生年月日',

`avatar` varchar(255) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '管理员头像',

PRIMARY KEY (`admin\_id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=2 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci ROW\_FORMAT=DYNAMIC;

--

-- Table structure for table `t\_cinema`

--

DROP TABLE IF EXISTS `t\_cinema`;

CREATE TABLE `t\_cinema` (

`cinema\_id` int unsigned NOT NULL AUTO\_INCREMENT COMMENT '影院id',

`cinema\_name` varchar(100) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '影院名称',

`cinema\_phone` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '影院电话',

`province` varchar(50) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '影院所在省份',

`city` varchar(50) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '影院所在市',

`county` varchar(50) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '影院所在区县',

`specified\_address` varchar(255) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '影院详细地址',

PRIMARY KEY (`cinema\_id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=15 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci ROW\_FORMAT=DYNAMIC;

--

-- Table structure for table `t\_comment`

--

DROP TABLE IF EXISTS `t\_comment`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

CREATE TABLE `t\_comment` (

`comment\_id` int unsigned NOT NULL AUTO\_INCREMENT COMMENT '评论id',

`user\_id` int unsigned DEFAULT NULL COMMENT '用户id',

`movie\_id` int unsigned DEFAULT NULL COMMENT '电影id',

`user\_score` int unsigned DEFAULT NULL COMMENT '用户评分',

`comment\_content` text CHARACTER SET utf8 COLLATE utf8\_unicode\_ci COMMENT '用户评论内容',

`comment\_date` datetime(6) DEFAULT NULL COMMENT '评论日期',

`support\_num` int unsigned DEFAULT NULL COMMENT '点赞数',

`is\_pass` int DEFAULT '0' COMMENT '评论是否通过审核',

`support\_user` text CHARACTER SET utf8 COLLATE utf8\_unicode\_ci COMMENT '点赞用户数组',

PRIMARY KEY (`comment\_id`) USING BTREE,

KEY `t\_comment\_ibfk\_1` (`user\_id`) USING BTREE,

KEY `t\_comment\_ibfk\_2` (`movie\_id`) USING BTREE,

CONSTRAINT `t\_comment\_ibfk\_1` FOREIGN KEY (`user\_id`) REFERENCES `t\_user` (`user\_id`) ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT `t\_comment\_ibfk\_2` FOREIGN KEY (`movie\_id`) REFERENCES `t\_movie` (`movie\_id`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=59 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci ROW\_FORMAT=DYNAMIC;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `t\_hall`

--

DROP TABLE IF EXISTS `t\_hall`;

CREATE TABLE `t\_hall` (

`hall\_id` int unsigned NOT NULL AUTO\_INCREMENT COMMENT '影厅id',

`cinema\_id` int unsigned DEFAULT NULL COMMENT '影院id',

`name` varchar(50) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '影厅名称',

PRIMARY KEY (`hall\_id`) USING BTREE,

KEY `t\_hall\_ibfk\_1` (`cinema\_id`) USING BTREE,

CONSTRAINT `t\_hall\_ibfk\_1` FOREIGN KEY (`cinema\_id`) REFERENCES `t\_cinema` (`cinema\_id`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=41 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci ROW\_FORMAT=DYNAMIC;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `t\_movie`

--

DROP TABLE IF EXISTS `t\_movie`;

CREATE TABLE `t\_movie` (

`movie\_id` int unsigned NOT NULL AUTO\_INCREMENT COMMENT '电影id',

`name` varchar(50) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '电影名称',

`poster` varchar(100) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '电影海报',

`director` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '导演',

`actor` varchar(100) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '主演',

`movie\_long` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '电影片长',

`language` varchar(50) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '电影语言版本',

`intro` text CHARACTER SET utf8 COLLATE utf8\_unicode\_ci COMMENT '电影简介',

`type` varchar(50) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '电影类型',

`public\_date` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '电影上映时间',

`wish\_num` int unsigned DEFAULT '0' COMMENT '想看人数',

`score` decimal(4,1) DEFAULT NULL COMMENT '电影综合评分',

PRIMARY KEY (`movie\_id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=39 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci ROW\_FORMAT=DYNAMIC;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `t\_order`

--

DROP TABLE IF EXISTS `t\_order`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!50503 SET character\_set\_client = utf8mb4 \*/;

CREATE TABLE `t\_order` (

`order\_id` int unsigned NOT NULL AUTO\_INCREMENT COMMENT '订单id',

`user\_id` int unsigned DEFAULT NULL COMMENT '用户id',

`schedule\_id` int unsigned DEFAULT NULL COMMENT '影院id',

`order\_phone` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '下单手机',

`order\_date` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '用户下单时间',

`ticket\_num` int unsigned DEFAULT NULL COMMENT '购买电影票数',

`ticket\_total\_price` decimal(10,2) unsigned DEFAULT NULL COMMENT '电影票单价',

`order\_seat\_info` varchar(100) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '座位信息',

`pay\_type` varchar(50) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '支付方式',

`phone\_code` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '手机取票码',

PRIMARY KEY (`order\_id`) USING BTREE,

KEY `t\_order\_ibfk\_1` (`user\_id`) USING BTREE,

KEY `t\_order\_ibfk\_2` (`schedule\_id`) USING BTREE,

CONSTRAINT `t\_order\_ibfk\_1` FOREIGN KEY (`user\_id`) REFERENCES `t\_user` (`user\_id`) ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT `t\_order\_ibfk\_2` FOREIGN KEY (`schedule\_id`) REFERENCES `t\_schedule` (`schedule\_id`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=102 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci ROW\_FORMAT=DYNAMIC;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `t\_schedule`

--

DROP TABLE IF EXISTS `t\_schedule`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!50503 SET character\_set\_client = utf8mb4 \*/;

CREATE TABLE `t\_schedule` (

`schedule\_id` int unsigned NOT NULL AUTO\_INCREMENT COMMENT '排片id',

`movie\_id` int unsigned DEFAULT NULL COMMENT '电影id',

`cinema\_id` int unsigned DEFAULT NULL COMMENT '影院id',

`hall\_name` varchar(50) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '影厅名字',

`show\_date` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '放映日期',

`show\_time` varchar(10) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '放映时间',

`price` decimal(10,2) DEFAULT NULL COMMENT '电影售价',

`seat\_info` text CHARACTER SET utf8 COLLATE utf8\_unicode\_ci COMMENT '座位信息',

PRIMARY KEY (`schedule\_id`) USING BTREE,

KEY `hall\_id` (`hall\_name`) USING BTREE,

KEY `t\_schedule\_ibfk\_1` (`movie\_id`) USING BTREE,

KEY `t\_schedule\_ibfk\_2` (`cinema\_id`) USING BTREE,

CONSTRAINT `t\_schedule\_ibfk\_1` FOREIGN KEY (`movie\_id`) REFERENCES `t\_movie` (`movie\_id`) ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT `t\_schedule\_ibfk\_2` FOREIGN KEY (`cinema\_id`) REFERENCES `t\_cinema` (`cinema\_id`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=102 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci ROW\_FORMAT=DYNAMIC;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `t\_user`

--

DROP TABLE IF EXISTS `t\_user`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!50503 SET character\_set\_client = utf8mb4 \*/;

CREATE TABLE `t\_user` (

`user\_id` int unsigned NOT NULL AUTO\_INCREMENT COMMENT '用户id',

`user\_name` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '用户名',

`avatar` varchar(100) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '用户头像',

`password` varchar(100) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '用户密码',

`phone` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '用户手机号码',

`sex` varchar(10) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '用户性别',

`birthday` varchar(20) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '用户出生年月日',

`sign` varchar(100) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '用户签名',

PRIMARY KEY (`user\_id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=55 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci ROW\_FORMAT=DYNAMIC;

/\*!40101 SET character\_set\_client = @saved\_cs\_client \*/;

--

-- Table structure for table `t\_wishmovie`

--

DROP TABLE IF EXISTS `t\_wishmovie`;

/\*!40101 SET @saved\_cs\_client = @@character\_set\_client \*/;

/\*!50503 SET character\_set\_client = utf8mb4 \*/;

CREATE TABLE `t\_wishmovie` (

`wishMovie\_id` int unsigned NOT NULL AUTO\_INCREMENT COMMENT '想看电影id',

`user\_id` int unsigned DEFAULT NULL COMMENT '用户id',

`movie\_id` int unsigned DEFAULT NULL COMMENT '电影id',

PRIMARY KEY (`wishMovie\_id`) USING BTREE,

KEY `t\_wishmovie\_ibfk\_1` (`user\_id`) USING BTREE,

KEY `t\_wishmovie\_ibfk\_2` (`movie\_id`) USING BTREE,

CONSTRAINT `t\_wishmovie\_ibfk\_1` FOREIGN KEY (`user\_id`) REFERENCES `t\_user` (`user\_id`) ON DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT `t\_wishmovie\_ibfk\_2` FOREIGN KEY (`movie\_id`) REFERENCES `t\_movie` (`movie\_id`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=119 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci ROW\_FORMAT=DYNAMIC;

-- Dump completed on 2021-05-26 19:37:39

## 7. 尚未解决的问题

目前阶段实现的界面还没有加入支付和验证等子系统，仅仅是电影售票管理的功能实现。

## 8.注解

B/S结构：B/S结构（Browser/Server，浏览器/服务器模式），是WEB兴起后的一种网络结构模式，WEB浏览器是客户端最主要的应用软件。这种模式统一了客户端，将系统功能实现的核心部分集中到服务器上，简化了系统的开发、维护和使用。客户机上只要安装一个浏览器，如Netscape Navigator或Internet Explorer，服务器安装SQL Server、Oracle、MYSQL等数据库。浏览器通过Web Server 同数据库进行数据交互。

My Sql：MySQL是一个精巧的SQL数据库管理系统，而且是开源的数据管理系统，是一个真正的多用户、多线程SQL数据库服务器。MySQL 主要目标是快速、健壮和易用。

Apache：Apache是世界使用排名第一的Web服务器软件。它可以运行在几乎所有广泛使用的计算机平台上，由于其跨平台和安全性被广泛使用，是最流行的Web服务器端软件之一。它快速、可靠并且可通过简单的API扩充，将Perl/Python等解释器编译到服务器中。同时Apache音译为阿帕奇，是北美印第安人的一个部落，叫阿帕奇族，在美国的西南部。也是一个基金会的名称、一种武装直升机等等。

Node.js：Node.js发布于2009年5月，由Ryan Dahl开发，是一个基于Chrome V8引擎的JavaScript运行环境，使用了一个事件驱动、非阻塞式I/O模型， 让JavaScript 运行在服务端的开发平台，它让JavaScript成为与PHP、Python、Perl、Ruby等服务端语言平起平坐的脚本语言。

Vue：Vue (读音 /vjuː/，类似于 view) 是一套用于构建用户界面的渐进式JavaScript框架。与其它大型框架不同的是，Vue 被设计为可以自底向上逐层应用。Vue 的核心库只关注视图层，不仅易于上手，还便于与第三方库或既有项目整合。另一方面，当与现代化的工具链以及各种支持类库结合使用时，Vue 也完全能够为复杂的单页应用（SPA）提供驱动

## 9、附录

暂无