

项目介绍

电影订票—QTZ私人影院

项目成员：吴潘研、刘缓缓、
简元宗、张莉

策划人：吴潘研

中心：杭州文教中心

2019年7月05日

目录

1 概述.....3

 1.1 产品概述及目目标.....3

 1.2背景介绍.....3

 1.3产品目的.....3

2 产品描述.....4

 2.0 团队架构.....4

 2.1 产品版本规划.....4

 2.2 产品需求描述.....6

 2.2 产品整体流程.....7

 2.3 数据库设计.....8

 2.4 界面面设计.....11

 2.5 功能交互设计.....12

 2.6 技术选型.....16

 2.7 开发周期.....17

 2.8 任务分配.....17

3 前端部署.....11

4 项目难点及解决.....18

5 项目4.0规划.....19

1 概述

1.1 产品概述及目目标

简述产品功能、预期实现目目标，可分阶段实现阶段性目目标。

本项目vue移动端-电影订票系统-QTZ私人影院APP，主要实现的功能为用户登录/注册、城市/影院选项页、电影列表、电影详情、订票模块，在线选座模块，商城小吃模块，购物车列表及支付功能等。计划工期为20天，共分为四个阶段，自2019年7月5日到2019年7月25日。

第一阶段：项目目规划设计，工期2天。

拟定设计方案，确定APP组成和功能，做准备工作，搜集资料，创建并完善数据库。

第二阶段：编码实现阶段，工期为7月8日到7月11日

将团队项目分工至个人，完成静态页面。

第三阶段：完成网页功能，工期为7月12日到7月22日。

将各个页面进行整合，逐步实现各项功能

第四阶段：项目测试与部署，工期为7月23日到7月26日

进行项目联调，完善项目功能，项目部署。

1.2背景介绍

随着互联网网的发展及升级,及时下单日日趋重要。电影订票APP是当前影城下单首选工具，有助于为客户提升下单效率、提升商家服务质量，优化商家服务结构，提升用户体验。未来电影订票系统服务将愈来愈快捷方便,日益科学、高效，即时下单将为城市居民带来更更舒适的体验感。

1.3产品目的

人民生活水平提高，消费能力提升，促进了电影订票蓬勃发展，人民更重视电影订票服务及效率。本产品电影订票APP——QTZ私人影院，主要提升商家服务质量，促进影城对于用户的整体验，加速发展。

电影订票APP作为商家影城与用户之间的重要媒介，能够更更快，更高效的传递消息，增加信息的公开程度，影片介绍结合图片详情，能够让用户有更直观的感受，更舒适的体验。让用户可以提前获悉影城动态及影片的详情，同时在进店后，即时下单，促进影城与客户之间的高沟通，使得商家为客户提供更优质

2 产品描述

2.0 团队架构

技术团队



组长：吴潘研
技术特点：产品、
全栈开发



组员：简元宗
技术特点：全栈
开发



组员：刘缓缓
技术特点：UI设
计、前端开发



组员：张莉
技术特点：UI设
计、前端开发

2.1 产品版本规划

版本	主要模块	功能点	计划开发时间	计划结束时间	备注
1.0	登录注册模块	用户根据规范格式注册账号，用户可从多个入口进入登录界面，登录成功后将登录信息存入localStorage	7月8日	7月10日	
	商品模块	展示商品信息，根据侧边栏切换商品类别	7月8日	7月10日	
	购物车模块	选择商品和数量加入购物车，根据单个商品数量动态计算出小计，再根据商品类别动态计算出总价，最终将购物信息存入vuex	7月8日	7月10日	

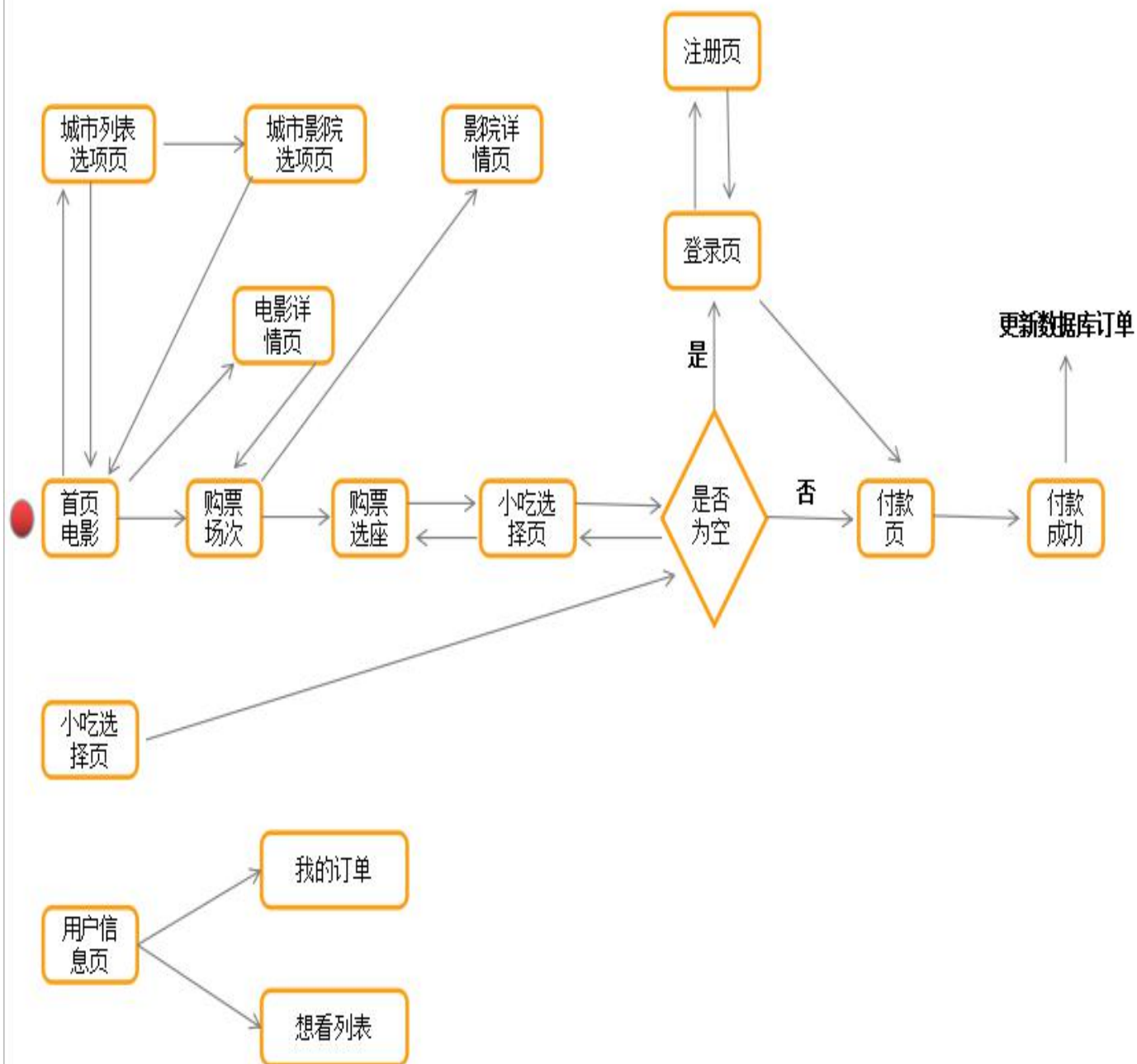
2.0	城市->影院选择	用户可自行选择城市和该城市中的影院，并提供了城市搜索功能（拼英搜索、汉字搜索、滑动锚点选择）	7月11日	7月20日	
	影院详情模块	影院环境展示（滑动/自动轮播图），信息展示、周边公交路线、地图及定位	7月11日	7月20日	
	电影列表模块（热映和即映）	根据上映状态分为热映和即映，可进入电影详情或直接购票	7月11日	7月20日	
	电影详情模块	电影简介、剧照、预告片，预售的电影用户可以加入想看列表，也可从当前页直接进入购票页	7月11日	7月20日	
	购票模块	根据用户选择的影院、电影进入购票页面，并可滑动电影栏选择电影（选中的电影在固定在中间位置且放大），然后选择电影场（日期/时间）次进行购票选座	7月11日	7月20日	
	选座模块	根据选择的场次进入选座，座位区域可移动、放大缩小，座位有可选、已售、已选、线上专享四种状态，用户可选择可选状态的座位并展示出座位坐标，也可通过再次点击已选座位或点击×取消选座，最多可选择6张	7月11日	7月20日	
	确认订单模块	用户选择电影座位和商品小吃后（可选），生成订单模板，点击确认订单进入支付界面（暂跳过），生成订单，进入用户订单模块	7月11日	7月20日	
	个人中心模块	展示用户的个人信息、用户订单列表、想看列表。进入订单列表可查看订单详情，进入想看列表可查看电影详情	7月11日	7月20日	

3.0	节流防抖	城市搜索添加防抖功能，城市锚点滑动添加防抖功能，首页滑动时头部吸附添加节流功能	7月21日	7月25日	
	页面缓存	当用户在各个页面间反复切换时，为减少重复操作和ajax请求次数，需将原来的页面状态缓存，如城市选择和影院选择，以及确认订单时判断登录状态，注册登录之后再跳回确认订单页面	7月21日	7月25日	
	使用webp格式图片	项目中用到的图片较多，加载时严重拖慢加载速度，将图片转换成webp格式，降低请求文件的大小，大幅提升加载速度，使用tomcat的web容器专门托管静态文件（图片、视频）	7月21日	7月25日	
	打包上线	将vue项目打包为静态文件放到服务端，将服务端从本地移植到阿里云服务器，并使用pm2启动服务，实现永久启动	7月21日	7月25日	

2.2 产品需求描述

本产品核心功能是为消费者提供订票系统，旨在为消费者之时，提供高效、高质量的服务质，提升服务体验及满意度。

2.2 产品整体流程



2.3数据库设计

2.3.1 数据库概述

数据库名：movie;

使用用数据环境：MYSQL

2.3.2 数据表清单

表名	描述	备注
user	用户表，记载注册后的用用户信息	基础信息，想看的电影记录
movies	电影表，记载电影的种类	电影详情页展示
cinemas	影城表，记载所有影城的详细信息关联到 电影表	影院选择
food	爆米花表，记载影城的爆米花种类	
orders	订单表，记载用户的订单，关联用户表	
carousel	轮播图片表	

movie orders old : int(11) # uid : int(11) # mid : int(11) # cid : int(11) date : varchar(32) time : varchar(32) # roomid : smallint(6) # price : int(11) choiceSeat : varchar(512) showSeat : varchar(512) productList : varchar(512)	movie carousel pid : int(11) pic : varchar(512)	movie foods fid : int(11) fName : varchar(12) fType : varchar(8) # sale : decimal(3,1) # price : decimal(3,1) # tid : int(11) fpic : varchar(64)	movie movies mid : int(11) mName : varchar(32) pic : varchar(128) score : char(3) intro : varchar(32) showTime : char(10) # watcher : int(11) # hot : tinyint(4) lanType : varchar(8) # duration : smallint(6) mType : varchar(8) mPhoto : varchar(512) story : varchar(512) director : varchar(16) actor : varchar(32) video : varchar(64) sType : varchar(64)	movie cinemas cid : int(11) # cityId : int(11) cName : varchar(32) loca : varchar(64) telephone : varchar(16) pic : varchar(512) map : varchar(64) bus : varchar(128) hall : varchar(32)
--	--	--	---	--

2.3.3数据表定义

用户信息表 (user)

字段名称	类型	备注	字段描述
uid	INT PRIMARY KEY AUTO_INCREMENT		用户的ID，为用户的唯一标识，由系统自动生成
uname	VARCHAR(16)		
phone	Char(11)		
sex	tinyint(4)		
upwd	VARCHAR(32)		
want	VARCHAR(128)		

电影表 (movies)

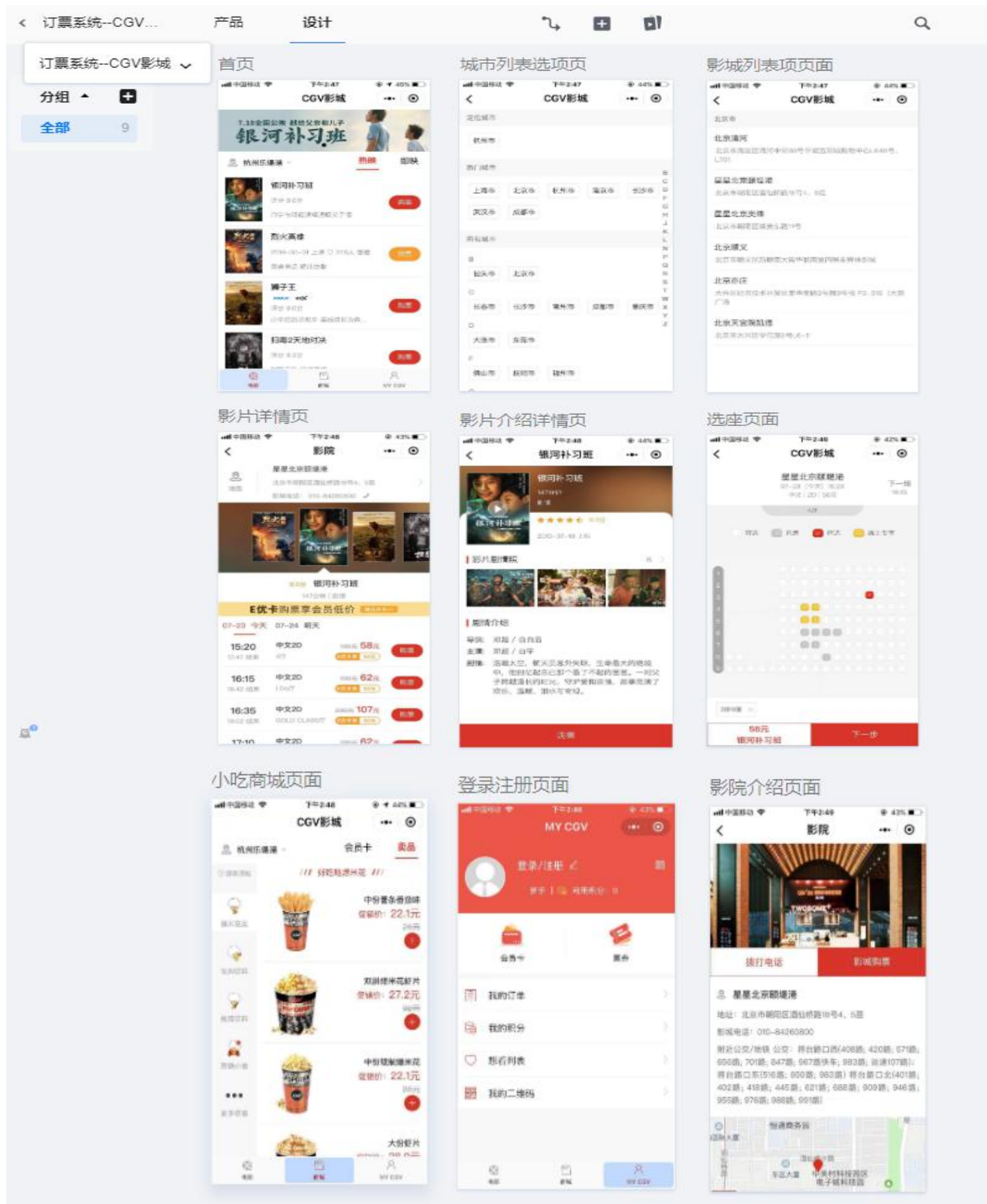
字段名称	类型	备注	字段描述
mid	INT PRIMARY KEY AUTO_INCREMENT	影片Id	电影的ID，为用电影片户的唯一标识，由系统自动生成
mName	VARCHAR(32)	影片名称	
pic	VARCHAR(128)	缩略图	
score	Char(3)	评分	
intro	VARCHAR(32)	短句简介	
showTime	Char (10)	上映时间	
watcher	Int(11)	想看人数	
hot	Tinyint(4)	热映 or 即映	
lanType	VARCHAR(8)	语言类型	
duration	smallint(6)	影片时长	
mType	VARCHAR(8)	影片类型	
mPhoto	VARCHAR(512)	剧照	
story	VARCHAR(512)	剧情简介	

director	VARCHAR(16)	导演	
actor	VARCHAR(32)	主演	
video	VARCHAR(64)	电影视频简介	
sType	VARCHAR(64)	播放类型	

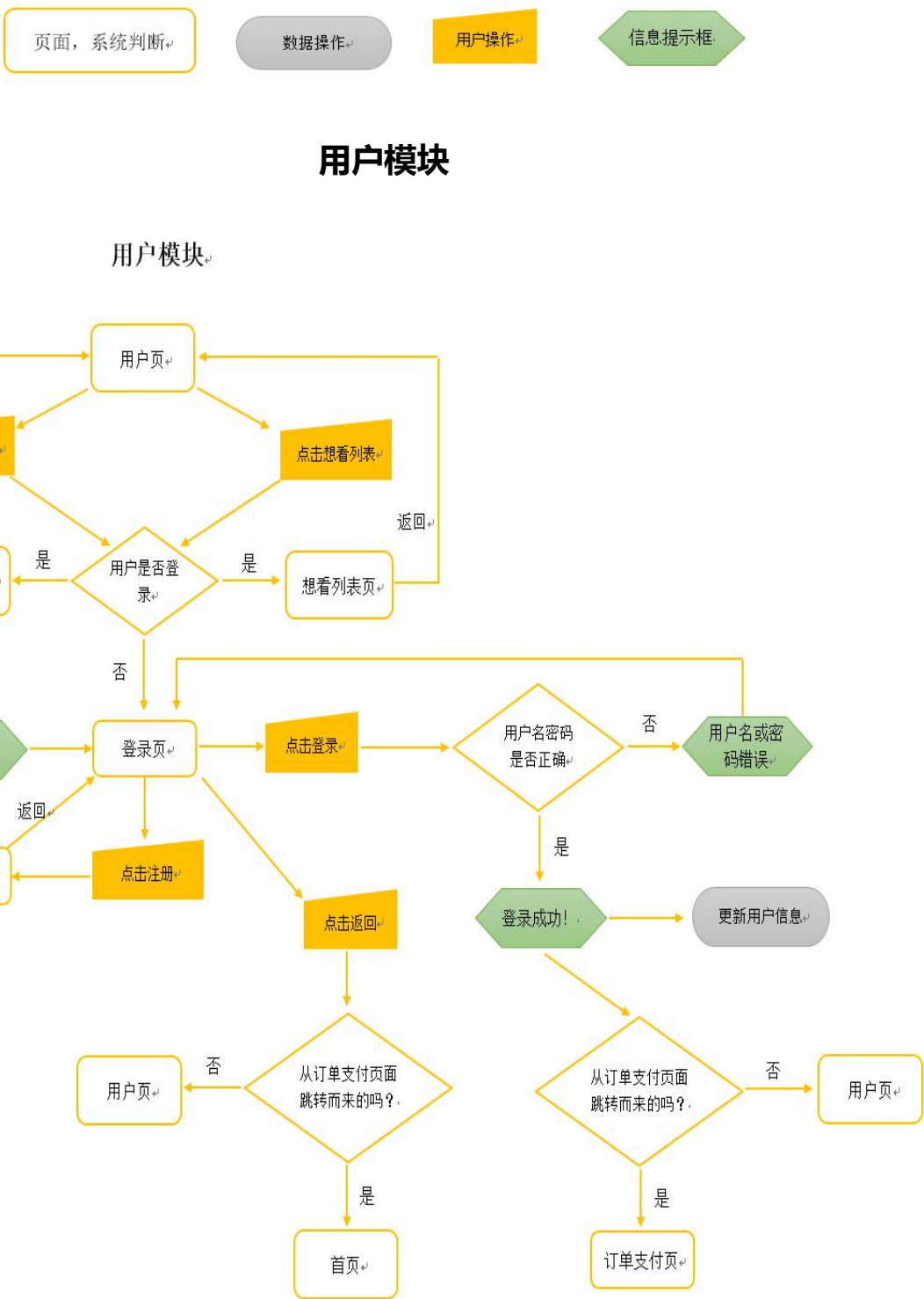
影城表 (cinemas)

字段名称	类型	备注	字段描述
cid	INT(11) INT PRIMARY KEY AUTO_INCREMENT	影院id	影院的ID，为影院的唯一标识，由系统自动生成
cityId	INT(11)	影院城市	
cName	VARCHAR(32)	影院名称	
loca	VARCHAR(64)	影院地址	
telephone	VARCHAR(16)	影院电话	
pic	VARCHAR(512)	影院照片	
map	VARCHAR(64)	影院地图	
bus	VARCHAR(128)	影院公交	
hall	VARCHAR(32)	影院影厅	

2.4 界面设计

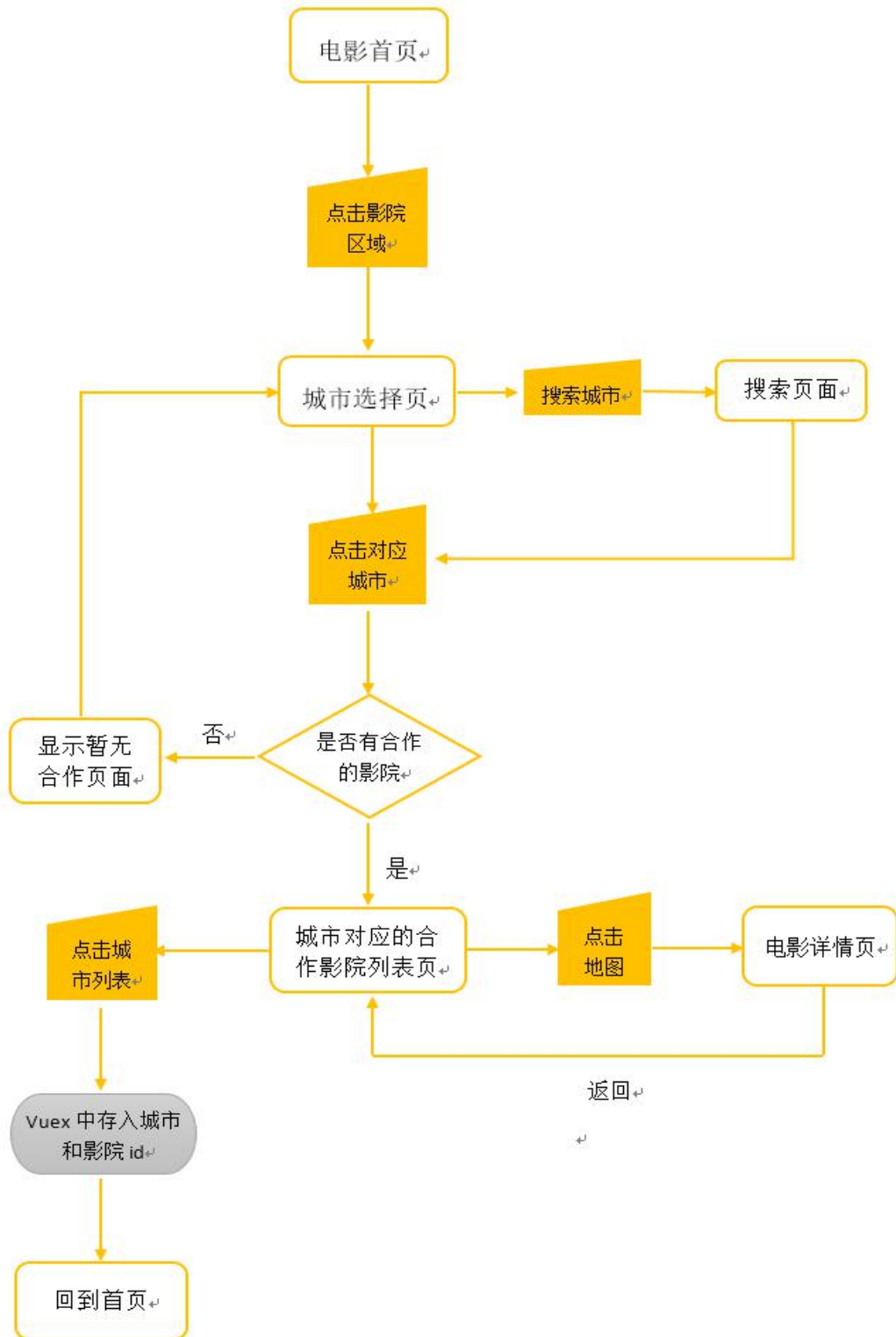


2.5 功能交互设计



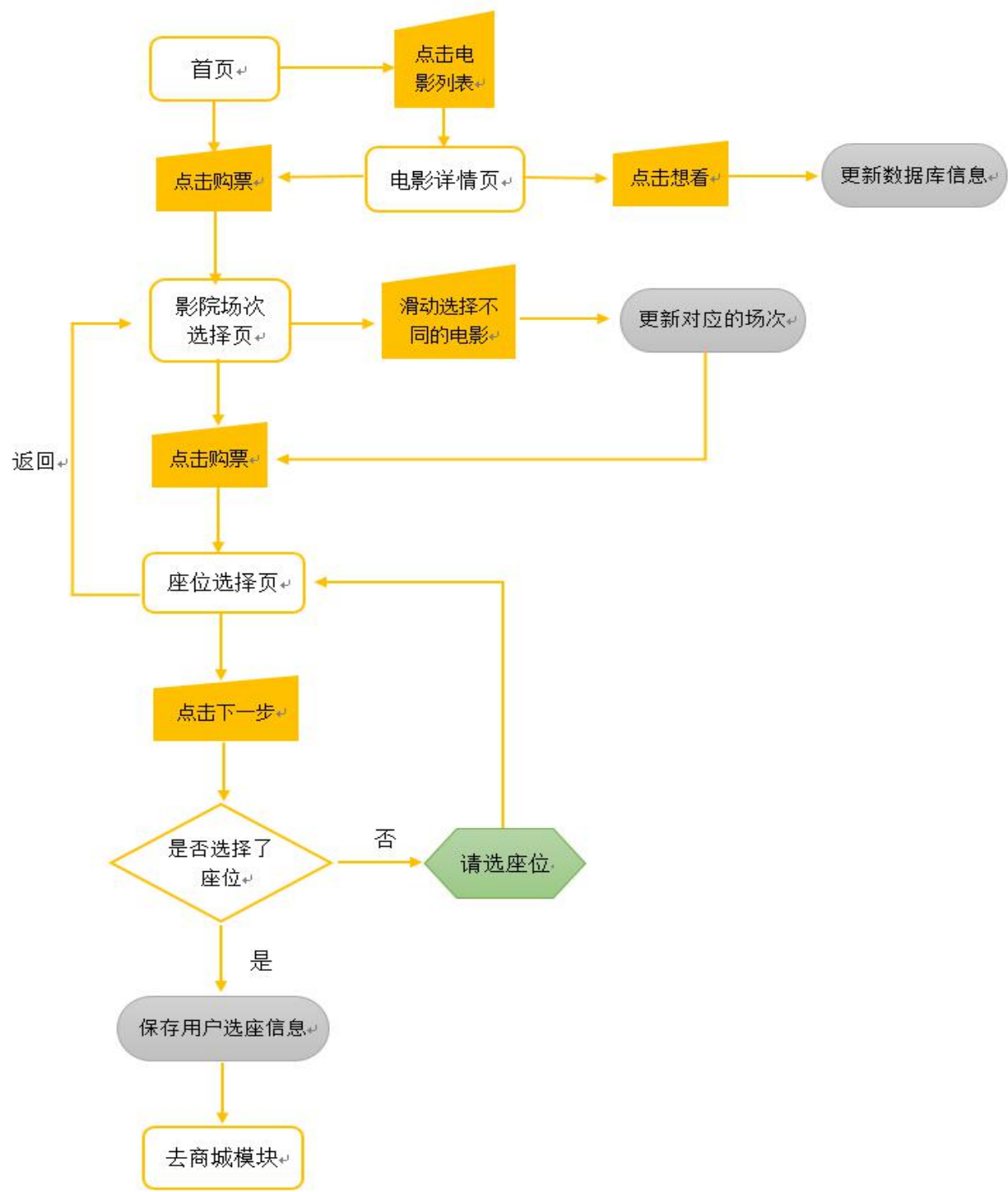
城市影院模块

城市影院模块



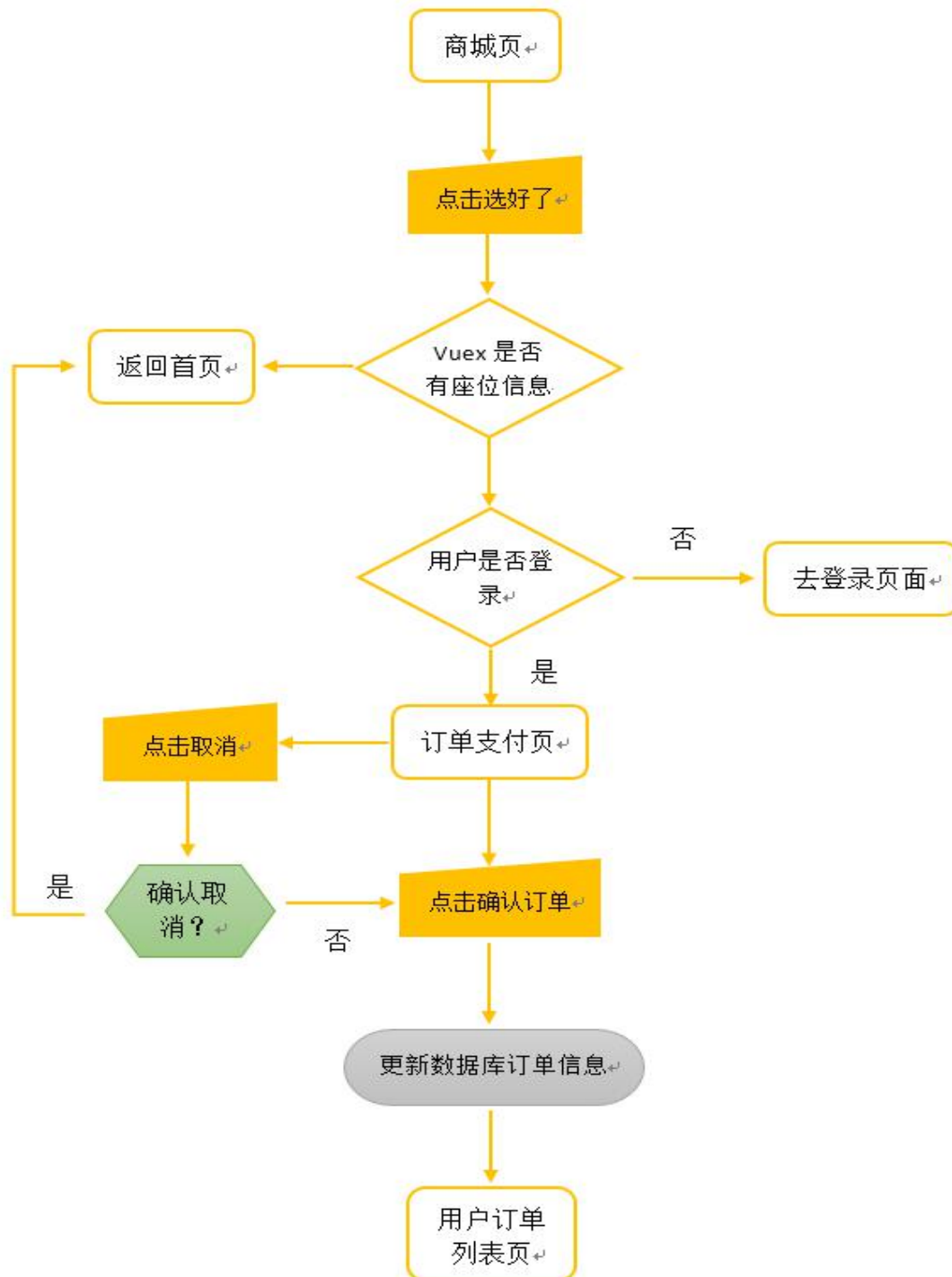
选座购票模块

选座购票模块

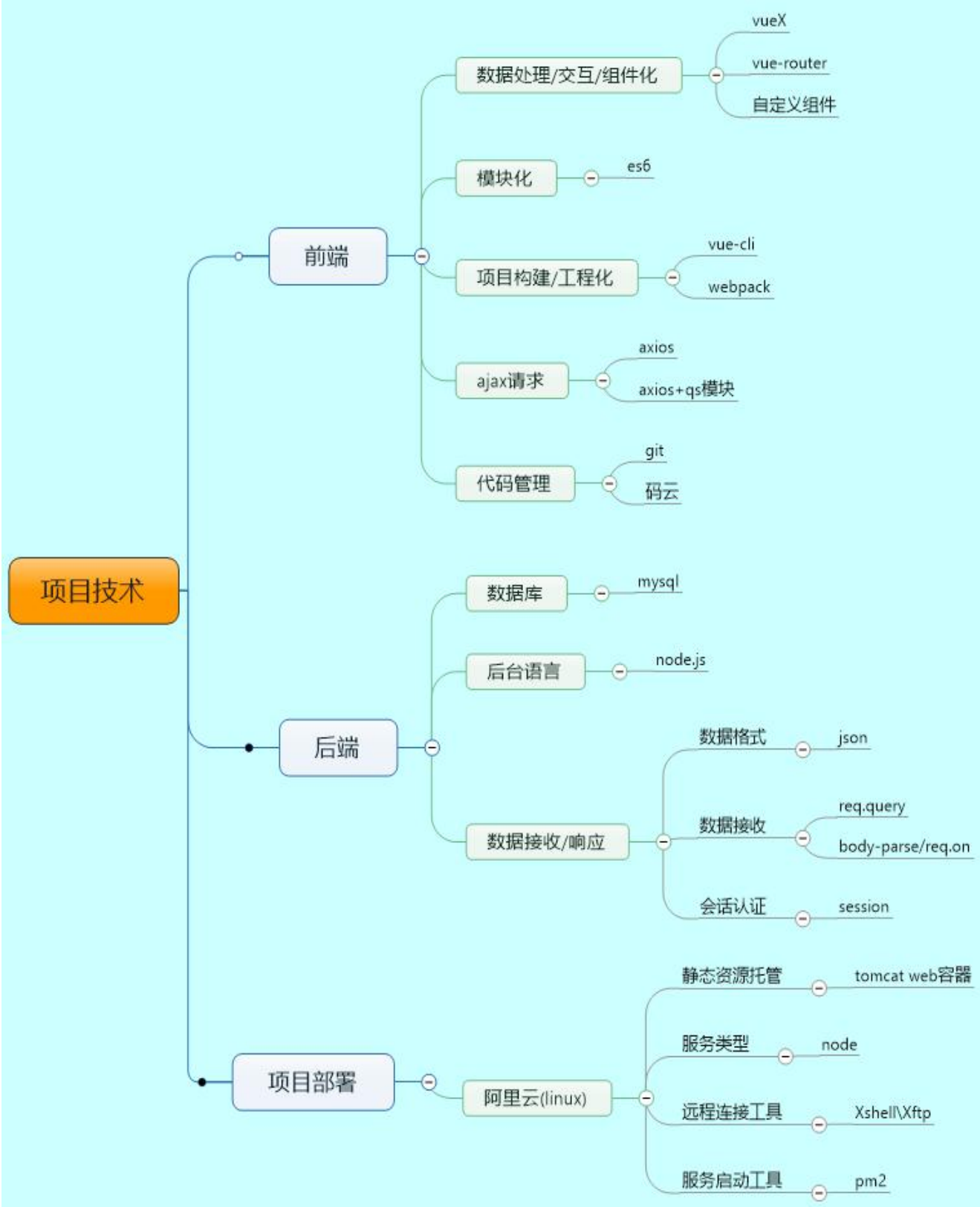


商城与订单模块

商城与订单模块



2.6 技术选型



2.7 开发周期

	开发阶段目标	任务内容	计划开发时间	计划结束时间	备注
阶段一	项目规划设计	前期素材的整理，功能点的梳理，mysql数据库的构建	2019.7.5	2019.7.8	吴潘研统筹
阶段二	编码实现阶段	编码实现，各页面静态完成，问题梳理及总结	2019.7.8	2019.7.12	吴潘研统筹
阶段三	完成网页功能	网页功能点完成，从用户体验角度改善功能	2019.7.12	2019.7.24	吴潘研统筹
阶段四	项目测试与部署	完善项目功能	2019.7.24	2019.7.25	吴潘研统筹
阶段五	项目整合及上传	代码上传及整合，确认	2019.7.25	2019.7.26	吴潘研统筹

2.8 任务分配

任务序号	主要模块	功能点	责任人	开发时间	计划结束时间	备注
1	Login.vue	页面功能实现+接口	张莉	2019.7.5	2019.7.12	
2	Home.vue	页面功能实现+接口	吴潘研	2019.7.5	2019.7.12	
4	Booking.vue	页面功能实现+接口	简元宗	2019.7.5	2019.7.12	
5	ProdoctList.vue	页面功能实现+接口	刘缓缓	2019.7.5	2019.7.12	
6	项目测试和项目整合	实现各界面之间的跳转，代码的完整性	吴潘研、简元宗	2019.7.15	2019.7.20	
7	项目上线和优化	项目上线和优化	吴潘研、简元宗	2019.7.21	2019.7.25	

3 项目部署

UI设计图：采用用蓝湖进行任务创建/创建任务清单

代码管理仓库：通过码云代码集中管理 代码链接：https://gitee.com/wpy2015/movie_ticket

代码部署上线地址：<http://qtzmovie.tanfp.cn>

4 项目难点及解决

难点1：选择影院时的缓存问题

问题描述：城市列表页（页A，下简称A），到对应的影院列表页（页B）实现页面刷新，影院列表页（页B）到影院详情页（页C）实现判断是否为同一个影院，是则走缓存，否则刷新。回退的时候，C->B，B显示缓存，B->A，A显示缓存。之前采用的方法是通过路由守卫判断路由跳转的页面，然后设置元标签的keepAlive属性来达到是否缓存组件的目的，B->C则在C的beforeRouteEnter钩子中判断两次的路径是否一样，来达到是否刷新的目的，刷新通过activated钩子函数。但是有BUG，A->B可以实现刷新，B->C也可以实现判断是否刷新，但是C->B的时候，显示的一直是第一次缓存的B页面，与需求不符。

解决方法：用keep-alive的include属性，结合vuex动态控制include达到动态管理缓存。即在keep-alive标签中使用include属性，值为vuex中保存一个需要被缓存组件的数组，然后在组件内的路由钩子函数中去动态改变数组，达到动态管理缓存，实现业务需求。

难点2：电影场次及座位的数据构思

问题描述：每个电影都有自己的独立场次，层级较深，数据量较大；影厅的座位不规则，而展现给用户的是直观的x排y座，且座位有不同的状态（可选、已售、已选、线上专享）；

解决方法：

场次数据构思：不同的影院、不同的电影都会有不同的场次安排，场次又需要根据日期和时间来排序展示，所以需单独创建一个场次表存放所有场次信息，然后根据影院、电影、日期、时间，逐级查询对应的场次，并筛选排序。

影厅座位数据构思：每个电影院会有自己的影厅，影厅的座位信息与电影票售卖状态无关，即只是单纯的座位信息，所以可单独创建一个影厅表存放影厅的座位信息，然后根据场次中存的影厅编号来查询座位信息。而每个座位的售票状态由订单表中已售的座位来决定；

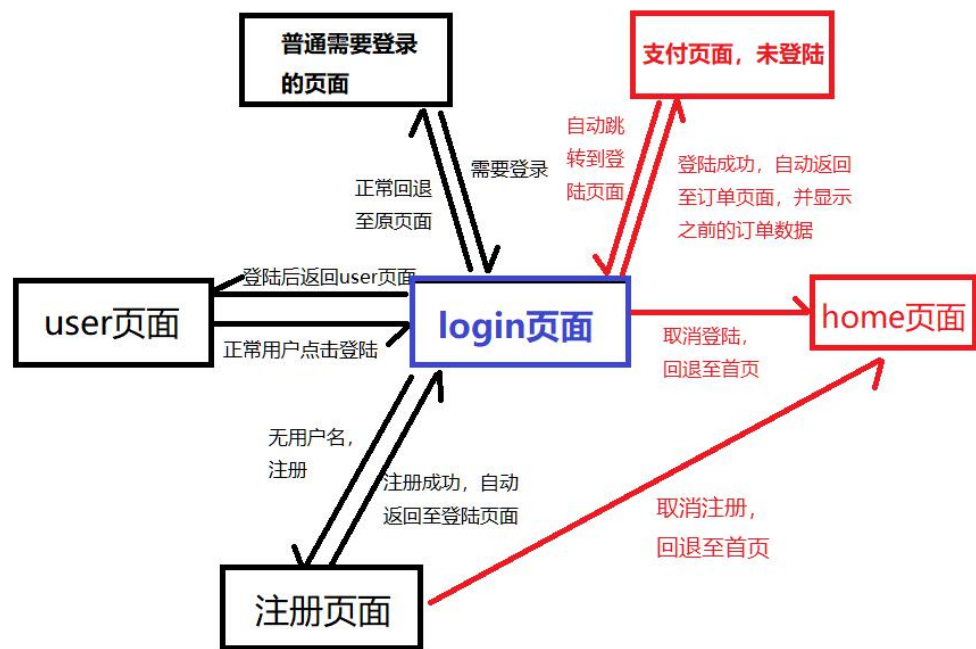
难点3：axios的post请求问题

问题描述：axios发送post请求默认提交的是json格式的数据，而在node中接收post请求的参数是通过body-parse（即querystring模块）中间件来转换参数的，导致参数接收不到，即req.body为空，或者数据格式不正确；

解决方法：在前端使用qs模块将对象转为字符串，以字符串方式提交给后端，后端依然使用body-parse处

理数据；也可不处理前端，在后端用原始的req.on监听请求，再用JSON.parse/JSON.stringify来处理数据；

难点4：用户Login页面，实现不同跳转，业务需求如下图



问题描述：可以看到，login页面有多个路由跳转和返回需求，其中从支付页面跳转来的页面，在登陆成功后要返回至订单页面，并且显示之前用户所选的订单信息。这里之前用了组件内的路由守卫，会有无限跳转和跳转混乱的问题。

解决方法：在Login路由元标签中存入一个属性jumpStaus，从支付页面跳转而来，则值设置为3，同时，页面的登陆，注册，回退按钮，也设置一个jumpStaus属性，用于判断是哪个按钮触发的操作，然后再加以判断，结合编程式导航进行跳转。凡事路由元标签中jumpStatus值为3的跳转都是带有支付参数的，登陆成功自动跳转到支付页面。支付完成后，清空这个参数，避免用户正常登陆跳转异常。没有此参数的，登陆成功后跳回用户主页面。在用户进行回退的时候，去判断路由是否带有支付参数，如果带有，则回退操作为取消支付，回退至首页，如果没有，则回退操作为正常操作，回退至上一页。

5 项目4.0规划

新增功能	功能描述和实现方式	开发时间	计划结束时间	备注
电影预告片	在加载速度有保障的情况下加入电影预告片	2019.7.30	2019.8.10	
订单过期功能	当用户购买的电影开场后，用户的订单过期；到达开场时间时，将超出时间范围的订单修改为过期状态	2019.7.30	2019.8.10	
场次禁售功能	当电影距离开场15分钟时，将该场次设为禁售状态，超出放映时间时删除该场次	2019.7.30	2019.8.10	
静态资源托管	将图片、音频、视频全部托管到阿里云的oss中，大幅提升加载速度	2019.7.30	2019.8.10	