中大实训中心	文档编号	233666www	版本	A1	密级	商密 A
	项目名称	布鲁电影				
	项目来源					

QR-RD-022(Ver1.2)

布鲁电影
详细设计说明书

（内部资料　请勿外传）

编　　写：	卞泽通	日 期：	2018.07.07
检　　查：	卞泽通	日 期：	2018.07.07
审　　核：	卞泽通	日 期：	2018.07.07
批　　准：	卞泽通	日 期：	2018.07.07

中大王青实训有限公司

文档变更记录

序号	变更（+/-）说明	作者	版本号	日期	批准
1	创建文档	卞泽通	V1.0	2018.07.07	卞泽通
2					

目　　录

1. 引言

1.1 编写目的和范围

本详细设计说明书编写的目的是说明程序模块的设计考虑，包括程序描述、输入/输出、算法和流程逻辑等，为软件编程和系统维护提供基础。本说明书的预期读者为系统设计人员、软件开发人员、软件测试人员和项目评审人员。

1.2 术语表

序号	术语或缩略语	说明性定义
1	锁定票	还未下单时锁定的座位
2		

1.3 参考资料

资料名称	作者	文件编号、版本	资料存放地点
微信小程序开发文档	*微信*	*最新版*	*网络*

1.4 使用的文字处理和绘图工具

文字处理软件：WPS
绘图工具：UMLet

2. 全局数据结构说明

2.1 常量

无。

2.2 变量

用户信息(string) userInfo
当前定位(string) currentLocation

当前选择的城市(string) currentCity

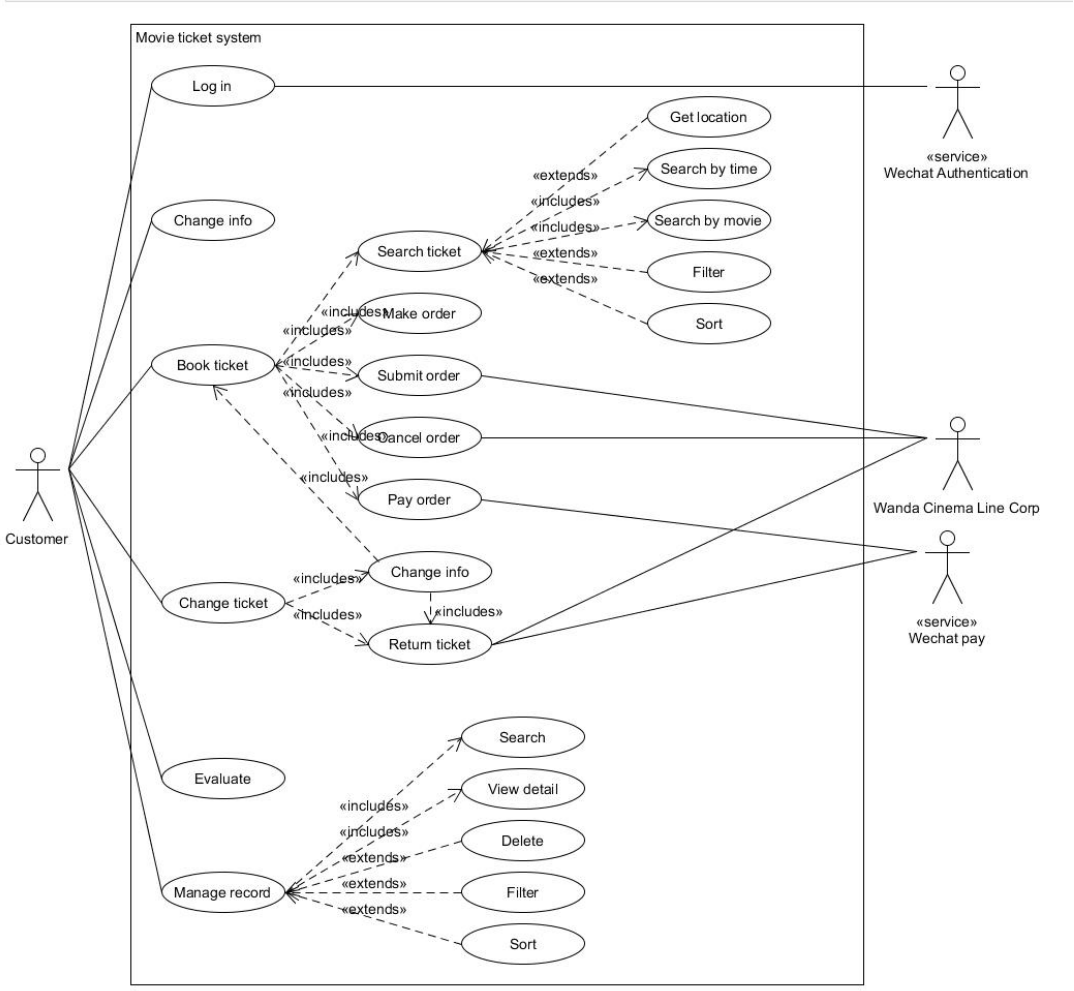
2.3 数据结构

无。

3. 模块设计

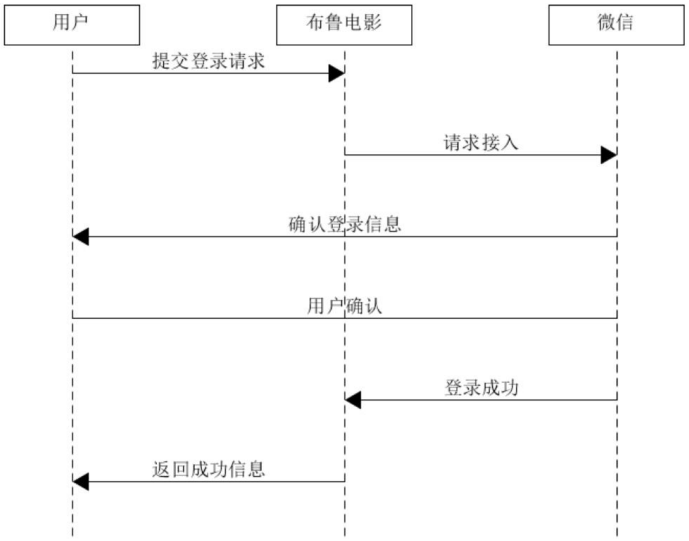
3.1 用例图

总用例图：

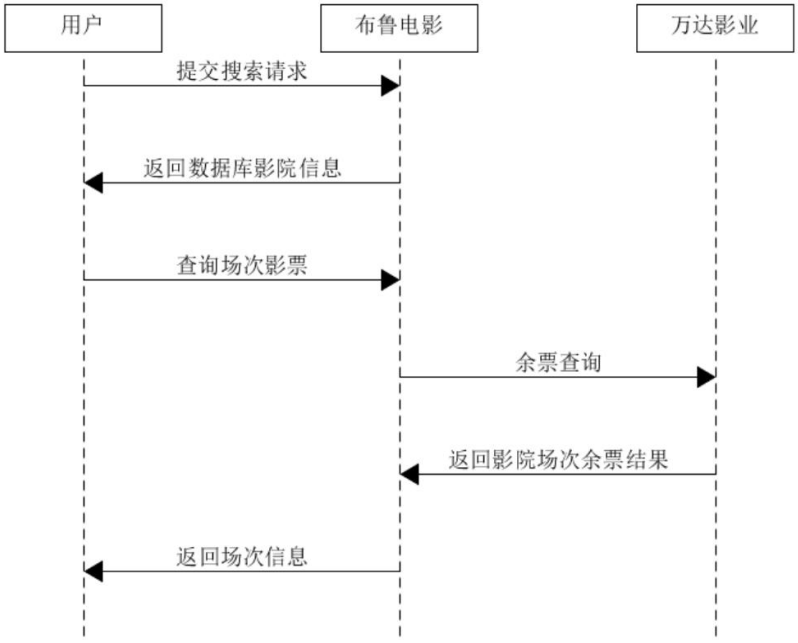


3.2 功能设计说明

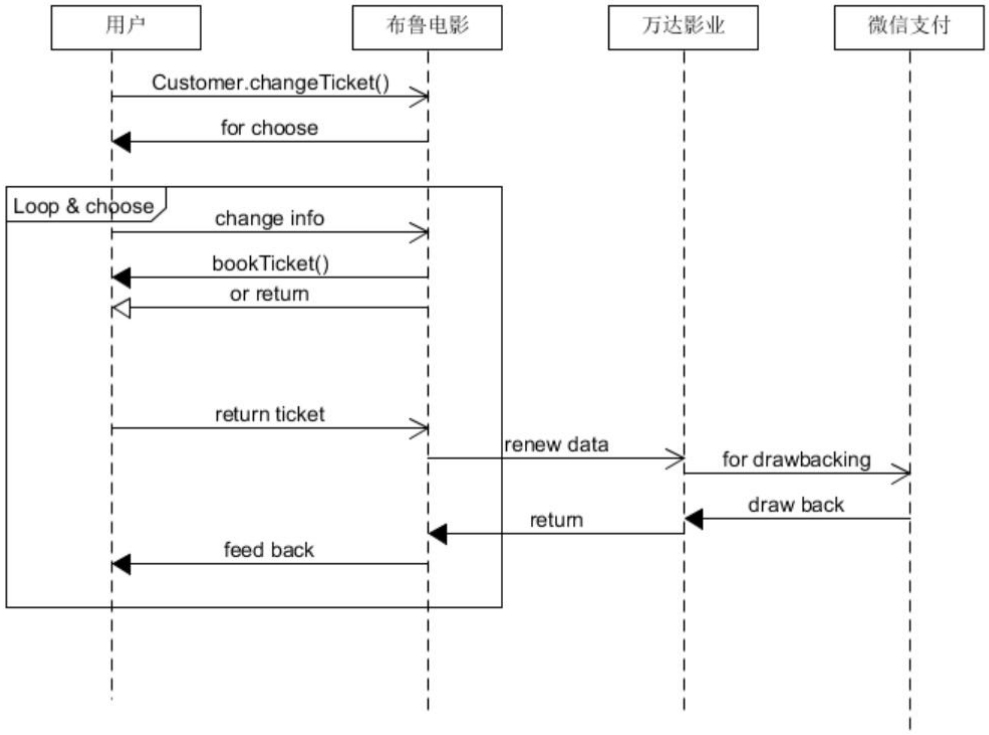
3.2.1 登录



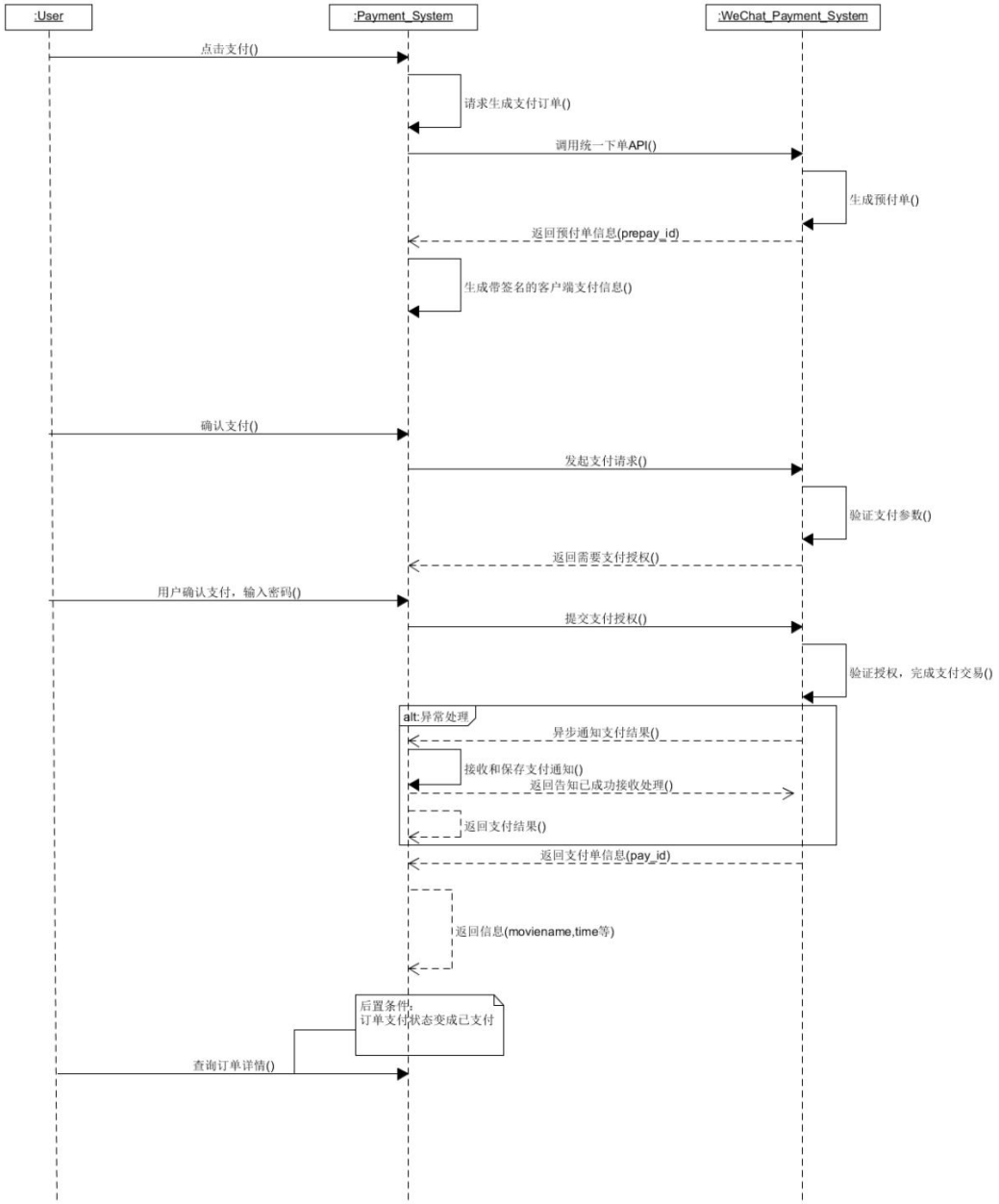
3.2.2 查询电影票



3.2.3 更换或取消电影票



3.2.4 支付


:User
:Payment_System
:WeChat_Payment_System
点击支付()
请求生成支付订单()
调用统一下单API()
生成预付单()
返回预付单信息(prepay_id)
生成带签名的客户端支付信息()
确认支付()
发起支付请求()
验证支付参数()
返回需要支付授权()
用户确认支付，输入密码()
提交支付授权()
验证授权，完成支付交易()
alt:异常处理
异步通知支付结果()
接收和保存支付通知()
返回告知已成功接收处理()
返回支付结果()
返回支付单信息(pay_id)
返回信息(moviename,time等)
后置条件:
订单支付状态变成已支付
查询订单详情()

4. 接口设计

4.1 用户接口

用户方面

用户注册

post user/regist

- data
  - code
  - encryptedData
  - iv
- return
  - resultCode
  - id
  - skey

用户登录

post user/login

- data
  - id
  - skey
- return
  - resultCode

4.2 信息接口

获取信息方面

获取影院信息

get /cinemas/get?aeraId=xxx&districtId=xxx&gps=2223488888&movieid=xxx

- return
  - resultCode
  - count
  - cinema array

获取电影信息

get /movie/get?cinemaid=xxx

- return
  - resultCode
  - count
  - movie array

获取场次信息

get / screenings/get?cinemaid=xxx&movieid=xxx&datetime=20180101

- return
  - resultCode
  - count
  - screening array

获取座位信息

get /seat/get?cinemaid=xxx&movieid=xxx&screeningsid=xxx

- return
  - resultCode
  - count
  - seats array

4.3 订单接口

订单方面

创建订单

post /order/create

- data
  - id
  - skey
  - cinemaId
  - movieId
  - screeningId
  - seatId
- return
  - resultCode
  - orderId

确认订单

post /order/confirm

- data
  - id
  - skey
  - orderId
- return
  - resultCode

支付订单

post /order/pay

- data
  - id
  - skey
  - orderId
  - pamentId
- return
  - resultCode

获取订单列表

post /order/getList

- data
  - id
  - skey
  - fromdate
  - todate
- return
  - resultCode
  - count
  - order array

获取订单信息

post /order/getOne

- data
  - id
  - skey
  - orderId
- return
  - resultCode
  - order data

取消订单

post /order/cancel

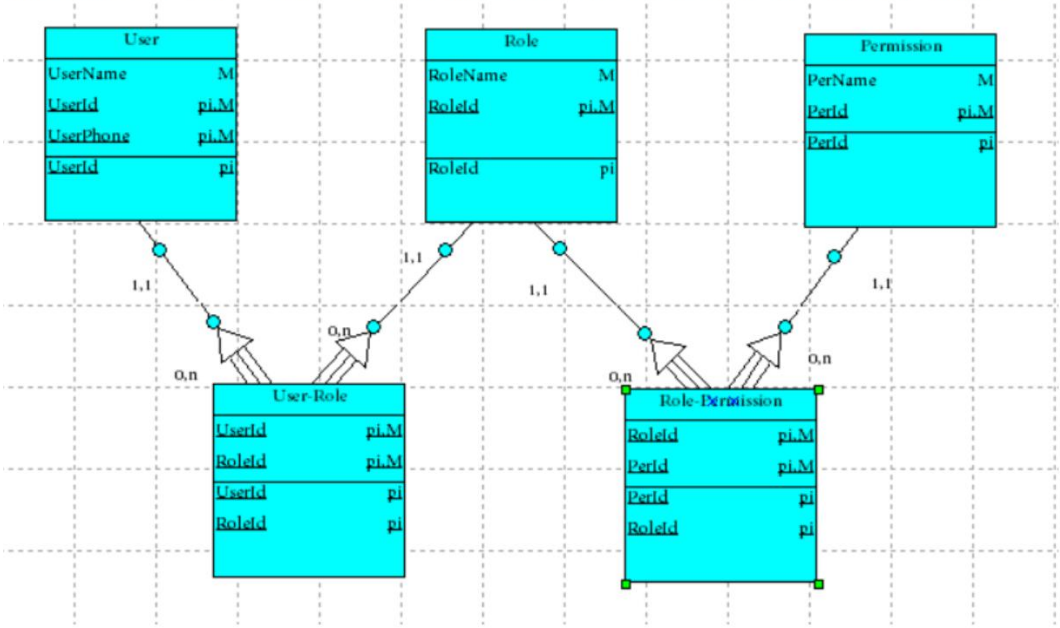
- data
  - id
  - skey
  - orderId
- return
  - resultCode

5. 数据库设计

5.1 用户及权限系统数据库设计

用户及权限系统数据库设计

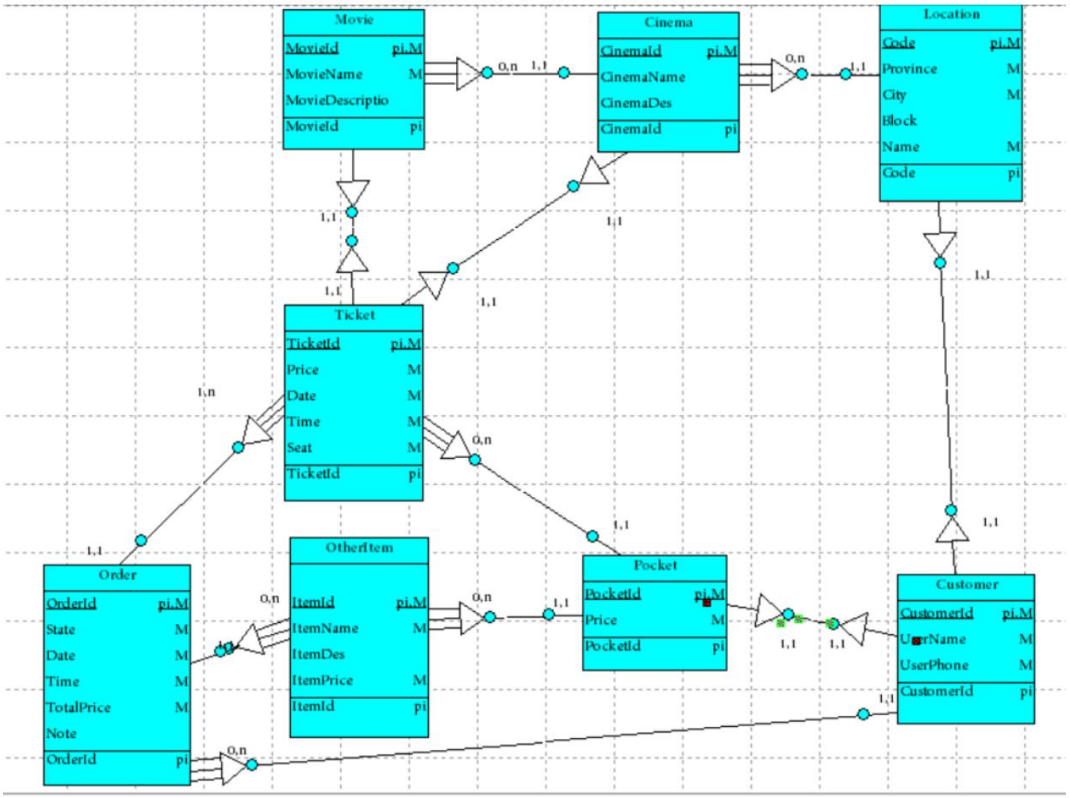
我们基于RBAC(Role-Based Access Control)做出了用户权限系统数据库的设计，用E-R图表示如下：



5.2 订票系统数据库设计

订票系统系统数据库设计

基于领域模型我们进行了整个订票系统的数据库设计，用E-R图表示如下：



6. 系统性能设计

6.1 数据访问层的设计时考虑系统性能

在设计方面，坚持高内聚、低耦合的原则，每一个控制模块只负责与自己有关的事物。

6.2 数据访问层的编程实现时考虑系统性能问题

6.2.1 避免查询大数据和大图片

将表按照功能尽可能地拆分，每次调用需要的数据，有效防止了大数据交互的情况发生。

6.2.2 尽可能减少查询数据量的大小

大多数用户可能不需要在有限的屏幕上看到庞大的数据，因此为了减少网络流量并且提高系统数据访问的性能，我们根据屏幕的最大显示数量减少了每次请求的数据量，在用户下拉界面时再请求新的数据，将数据大小介绍到可管理的范围之内。

6.3 数据库表设计时考虑系统性能问题

6.3.1 选择正确的数据类型

在设计数据库表结构时，我们尽可能地选择了效率最高的数据类型，在能用整型时避免使用浮点数，避免了许多不必要的性能损失。

6.3.2 对数据库进行最优配置

我们的数据表设计按照第三范式的要求，基本上保证了数据的完整性。

6.3.3 在经常被查询的数据库表字段中设计索引

我们通过设立 BTREE、设定 COLLATE 等方式建立了索引，方便查询。

```
`session_key` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL,
`user_info` varchar(2048) COLLATE utf8mb4_unicode_ci NOT NULL,
PRIMARY KEY (`open_id`),
KEY `openid` (`open_id`) USING BTREE,
KEY `skey` (`skey`) USING BTREE
```

7. 系统出错处理

会 log 出来，给一个反馈，方便调试修改。