

SHANGHAI JIAO TONG UNIVERSITY

**过程建模及优化作业**



业务名称： 基于手机的智能停车场系统

小组成员: 赵铭心（119037910064）

小组成员: 秦皓喆（119037910051）

小组成员: 杨震旦（119037910035）

目录

[基于手机的智能停车场系统 3](#_Toc28170121)

[1. 流程模型 3](#_Toc28170122)

[2. 实体交互图 4](#_Toc28170123)

[2.1实体交互图1 4](#_Toc28170124)

[2.3 实体交互图2 5](#_Toc28170125)

[3. 用例图 6](#_Toc28170126)

[4. 类图 11](#_Toc28170127)

[5. 数据流图 12](#_Toc28170128)

[6. 状态图 13](#_Toc28170129)

[6.1 状态图1 13](#_Toc28170130)

[6.2 状态图2 13](#_Toc28170131)

[7. 核心算法 14](#_Toc28170132)

[8. Petri网 15](#_Toc28170133)

[8.1 Petri网仿真 16](#_Toc28170134)

[8.3 结果分析 16](#_Toc28170135)

[9. 分工情况 17](#_Toc28170136)

# 基于手机的智能停车场系统

## 流程模型

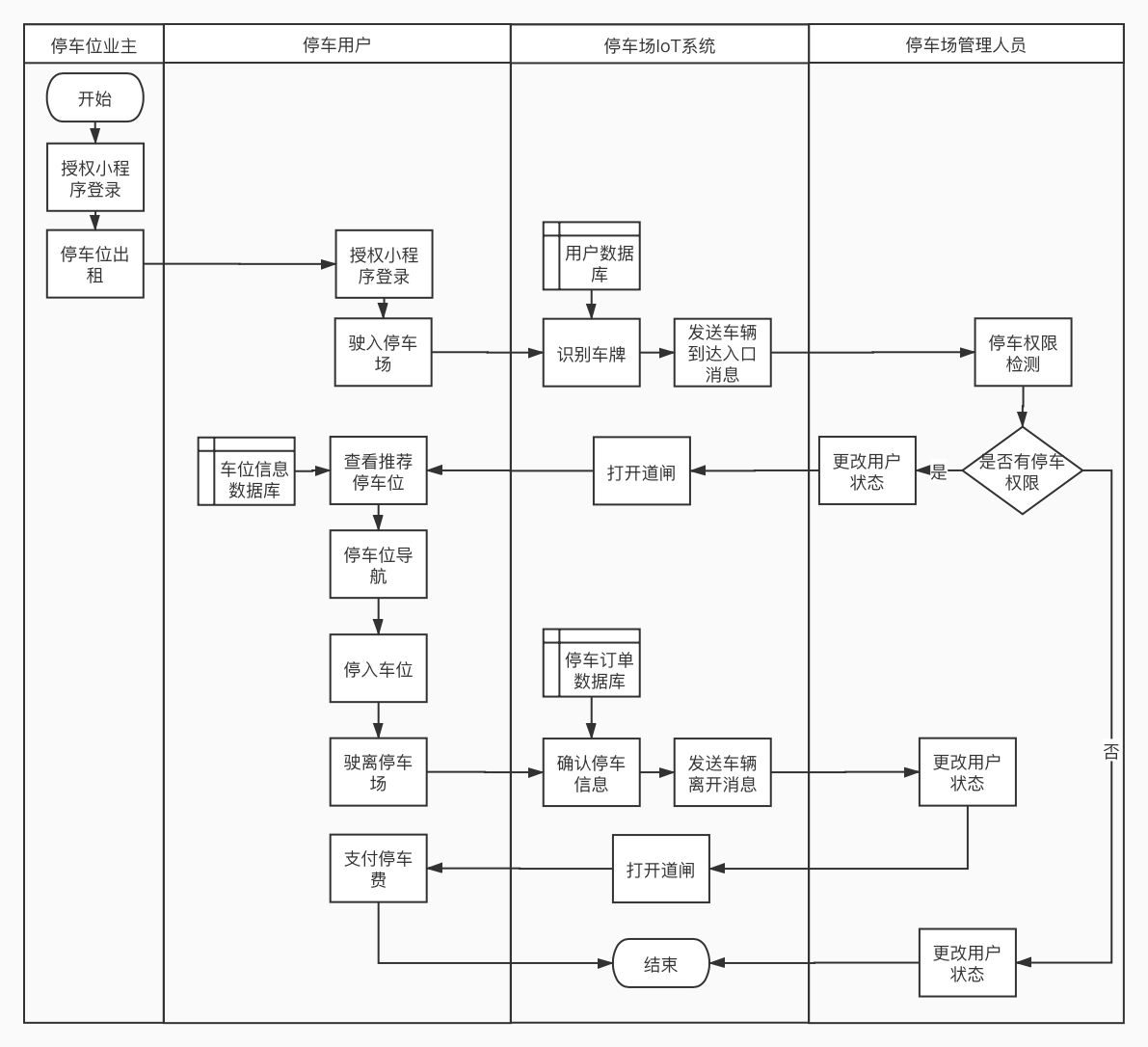


图1-1 流程图

停车位业主授权登录停车场小程序，进行车位出租。停车用户授权登陆小程序，车辆驶入停车场。IoT系统识别车牌并发送车辆到达入口信息到后台，后台进行用户权限检测。如果用户为黑名单用户，用户无法进入停车场流程结束。如果用户有权限进入则打开道闸，查看车位推荐，停车位导航并停入车位。之后用户驶离停车场，IoT系统确认用户离开，打开道闸，用户支付车费，流程结束。

## 实体交互图

### 2.1实体交互图1

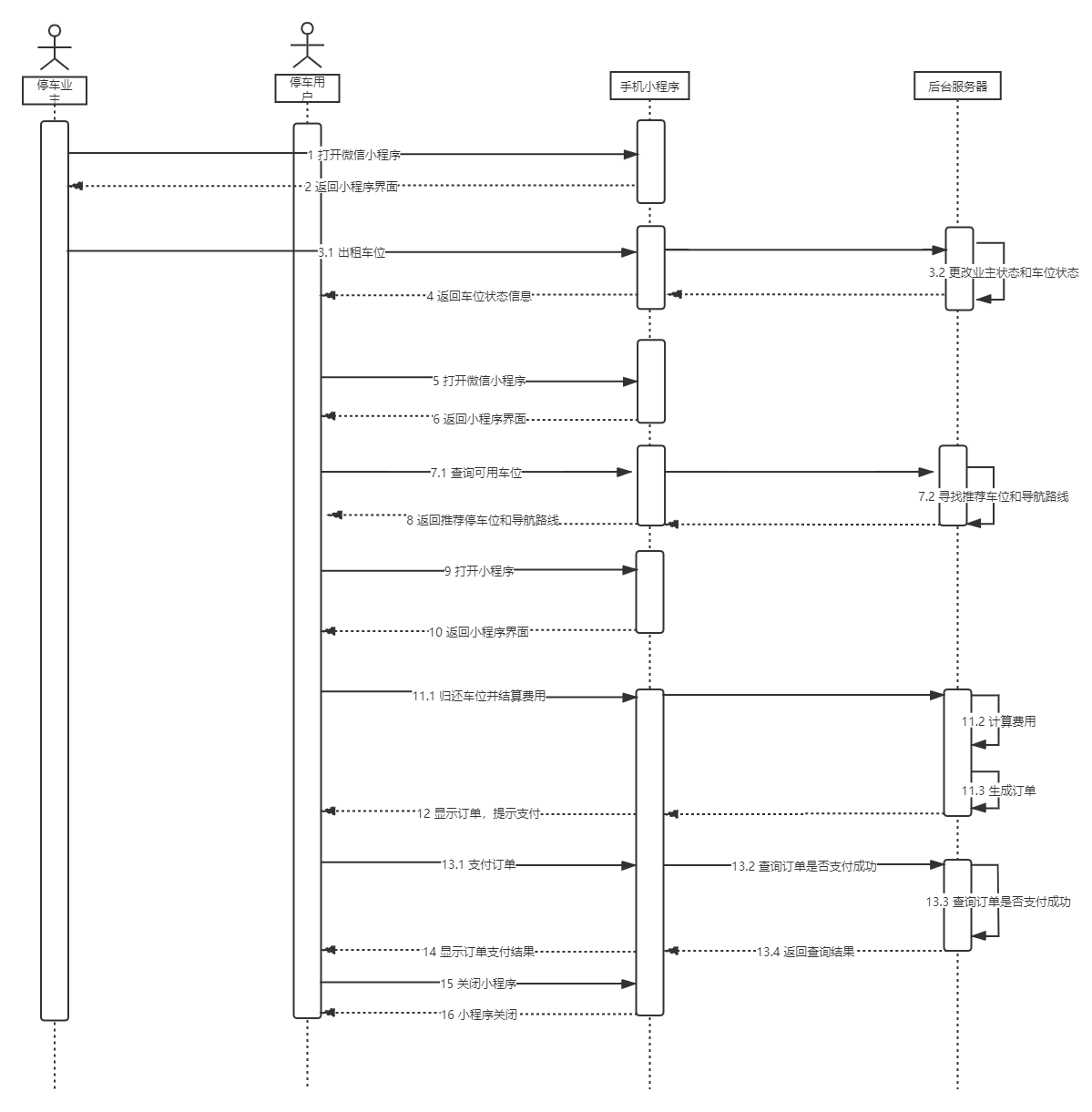


图2-1 实体交互图1

停车位业主打开小程序后申点击车位出租，小程序和服务器通讯修改业主车位的状态为已出租，并通过小程序告知业主车位出租情况。用户在进入停车场后打开小程序，查询可用车位。小程序将用户的请求发送到后台服务器后，后台服务器根据算法寻找推荐停车位并给出导航路线。用户归还车位并结算费用时，由后台服务器计算费用并生成订单，并在小程序端显示订单。用户支付订单后，小程序询问服务器订单是否支付成功并提示用户支付结果。用户关闭小程序。

### 2.3 实体交互图2

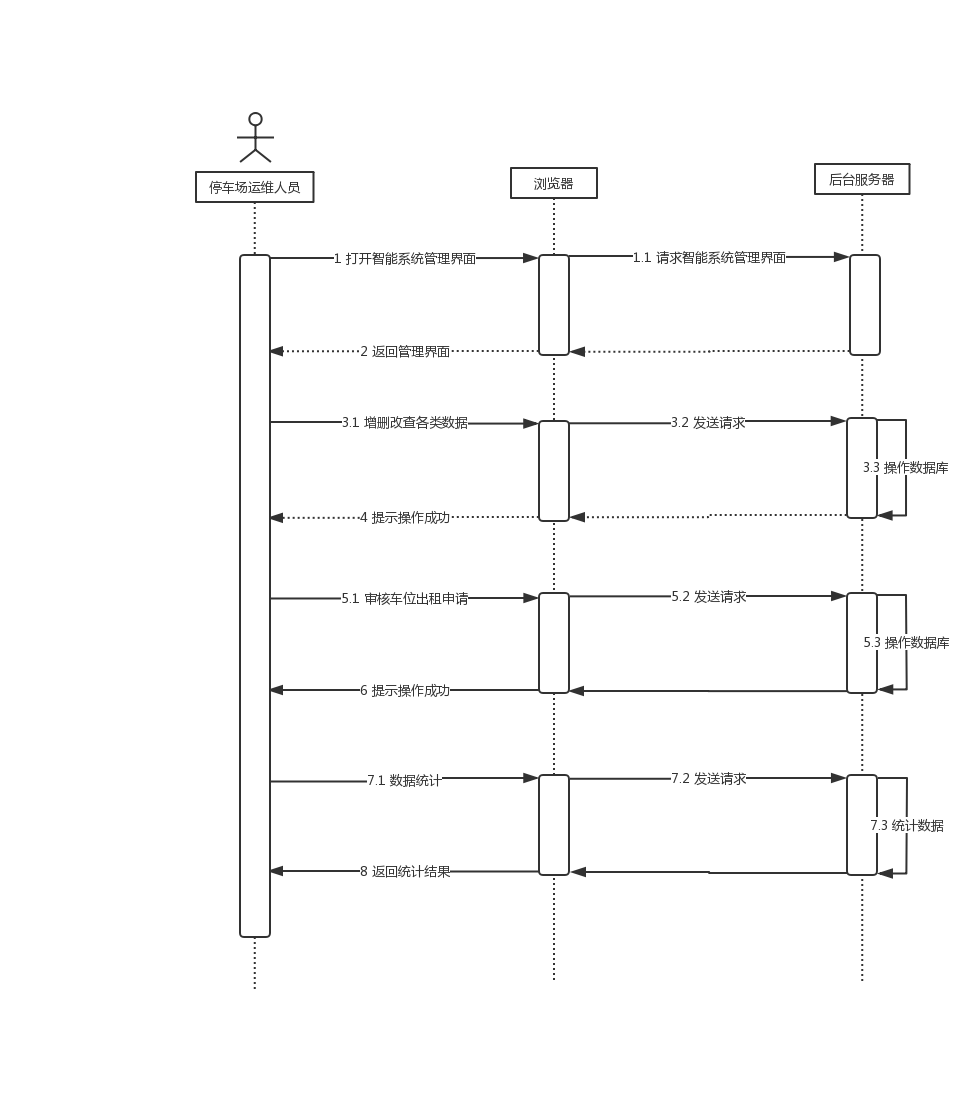


图2-2 实体交互图2

停车场运维人员登录智能停车场系统后台管理的界面，并通过服务器前端的接口向后端发送数据增删改查的请求。后端接收到前端请求后，通过调用数据库接口操作数据库。运维人员还可以通过前端提供的接口审核车位出租申请和进行各类数据的统计。操作的结果由后端处理后通过前端显示给运维人员

## 用例图

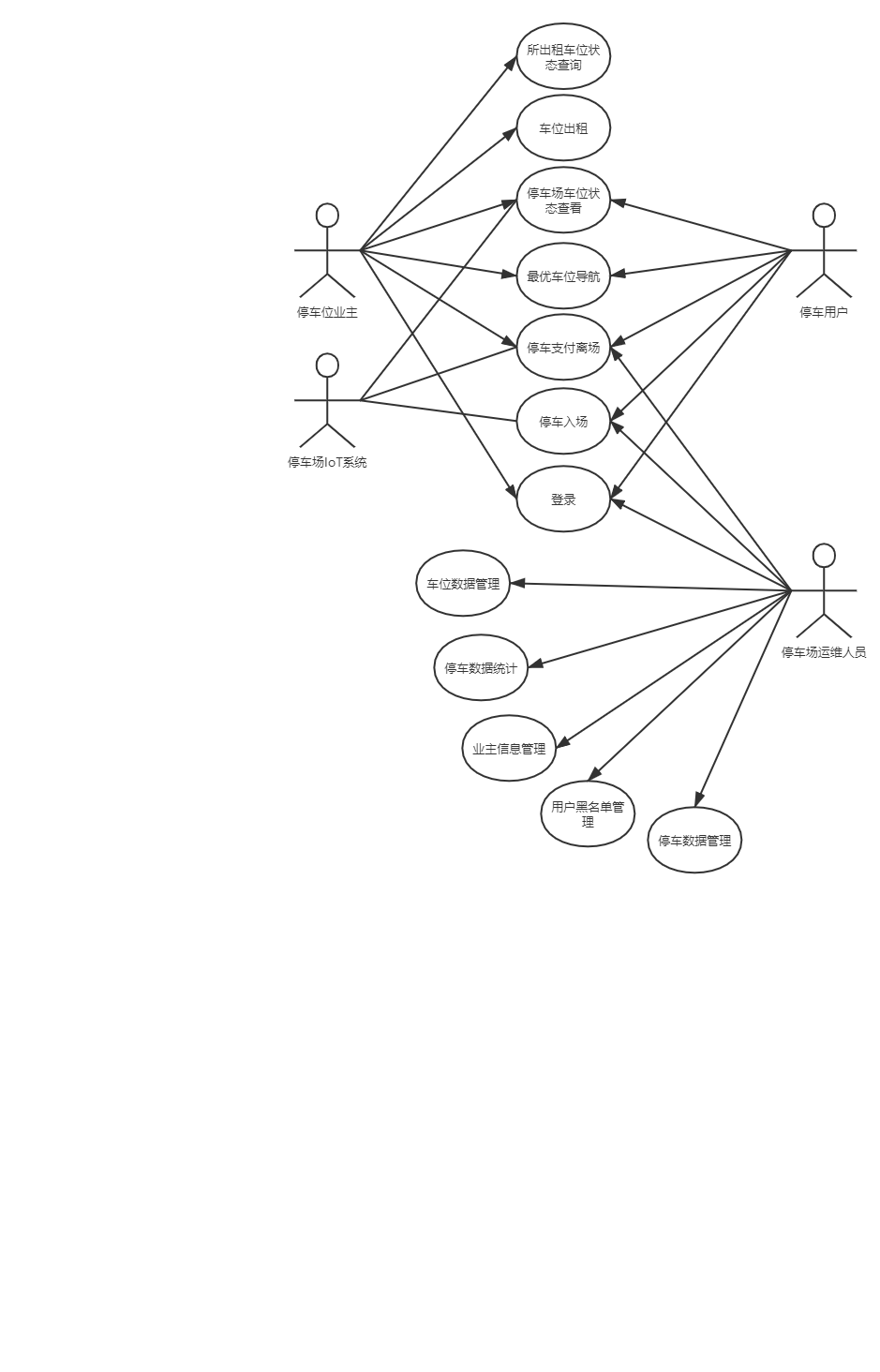


图3-1 用例图

功能清单如下：

|  |  |
| --- | --- |
| 用例名 | 所出租车位状态查询 |
| 描述 | 停车场业主查看所出租车位是否被占用 |
| 执行者 | 停车场业主 |
| 前置条件 | 业主授予了小程序权限进行身份识别并打开小程序 |
| 后置条件 | 无 |
| 基本流 | 1. 用户进入出租车位状态查询界面 2. 服务器返回用户所出租车位当前状态 3. 用户界面显示出租车位当前状态 |
| 备选流 | 无 |

|  |  |
| --- | --- |
| 用例名 | 车位出租 |
| 描述 | 停车位业主将自己的车位出租 |
| 执行者 | 停车场业主 |
| 前置条件 | 业主授予了小程序权限进行身份识别并打开小程序 |
| 后置条件 | 无 |
| 基本流 | 1. 用户进入出租车位的功能界面 2. 用户点击“出租”按钮 3. 服务器接受请求，并将出租状态存入数据库 4. 用户界面显示“车位已出租” |
| 备选流 | 1a.用户车位已全部出租。提示“无可出租车位”。 |

|  |  |
| --- | --- |
| 用例名 | 最优车位导航 |
| 描述 | 将停车用户及停车场业主导航到最优车位 |
| 执行者 | 停车用户、停车场业主 |
| 前置条件 | 用户授予了小程序权限进行身份识别并打开小程序 |
| 后置条件 | 无 |
| 基本流 | 1. 用户进入停车场并打开小程序 2. 服务器经过计算后返回最佳车位位置并进行导航 3. 用户跟随导航到达车位后，小程序显示“感谢您选择Peter Park” 4. 服务器收到停车位置信息，并更新停车场状态。 |
| 备选流 | 1a.小程序检测用户身份为业主且其车位未出租，服务器返回用户车位位置并为其进行导航。用户跟随导航到达车位后，小程序显示“感谢您选择Peter Park”  5a.用户没有跟随导航，而是选择了其他车位停车。服务器收到停车位置信息，并更新停车场状态。 |

|  |  |
| --- | --- |
| 用例名 | 停车支付离场 |
| 描述 | 停车后支付相关费用并离开停车场 |
| 执行者 | 停车场业主、停车用户 |
| 前置条件 | 用户授予了小程序权限进行身份识别并打开小程序 |
| 后置条件 | 小程序返回首页 |
| 基本流 | 1. 用户到达出口道闸附近并打开小程序进入订单页面，点击支付按钮 2. 服务器接受请求并返回所需支付金额 3. 用户点击支付 4. 服务器接受到支付信息，发送消息停车场IoT系统,使其打开道闸 |
| 备选流 | 1a．用户已绑定支付账户，到达道闸前后，服务器自动扣款并打开道闸。  3a.用户支付失败，返回步骤3等待用户重新进行支付操作 |

|  |  |
| --- | --- |
| 用例名 | 停车场车位状态查看 |
| 描述 | 停车用户或业主查看停车场当前车位状态信息 |
| 执行者 | 停车用户、停车位业主 |
| 前置条件 | 用户授予了小程序权限进行身份识别并打开小程序 |
| 后置条件 | 无 |
| 基本流 | 1. 用户进入查看车位状态的页面 2. 服务器返回当前停车场车位状态信息 |
| 备选流 | 无 |

|  |  |
| --- | --- |
| 用例名 | 停车入场 |
| 描述 | 用户进入停车场 |
| 执行者 | 停车用户、停车位业主 |
| 前置条件 | 用户将车辆行驶到道闸前 |
| 后置条件 | 无 |
| 基本流 | 1. IoT系统识别用户车牌并上传到服务器 2. 服务器检测用户具有停车权限，且停车场无空位 3. 服务器发送消息停车场IoT系统,使其打开道闸 |
| 备选流 | 2a.服务器检测到用户不具有停车权限，发送消息给停车场IoT系统，使其提醒用户离开  2b.停车场无空位，服务器发送消息给停车场IoT系统，使其提醒用户停车场已满 |

|  |  |
| --- | --- |
| 用例名 | 登录 |
| 描述 | 人员登入系统 |
| 执行者 | 停车场运维人员、业主、停车用户 |
| 前置条件 | 停车场运维人员、业主、停车用户登录系统 |
| 后置条件 | 页面跳转 |
| 基本流 | 1. 停车场运维人员打开停车场后台管理系统，并输入用户名和密码或业主、停车用户打开小程序并进行登录 2. 服务器接受到用户名和密码并审核通过 |
| 备选流 | 2a.服务器审核不通过，返回步骤1 |

|  |  |
| --- | --- |
| 用例名 | 停车数据管理 |
| 描述 | 停车场运维人员对停车数据进行增删改查 |
| 执行者 | 停车场运维人员 |
| 前置条件 | 停车场运维人员登录系统 |
| 后置条件 | 无 |
| 基本流 | 1. 停车场运维人员点击“停车数据管理” 2. 服务器返回停车数据 3. 停车场运维人员对数据进行操作并点击“保存” 4. 服务器保存用户所做的修改 |
| 备选流 | 3a.用户未点击“保存”，直接退出系统，流程结束 |

|  |  |
| --- | --- |
| 用例名 | 车位数据管理 |
| 描述 | 停车场运维人员进行车位数据增删改查 |
| 执行者 | 停车场运维人员 |
| 前置条件 | 停车场运维人员登录系统 |
| 后置条件 | 无 |
| 基本流 | 1. 停车场运维人员点击“车位数据管理” 2. 服务器返回车位数据 3. 停车场运维人员对数据进行操作并点击“保存” 4. 服务器保存用户所做的修改 |
| 备选流 | 3a.用户未点击“保存”，直接退出系统，流程结束 |

|  |  |
| --- | --- |
| 用例名 | 停车数据统计 |
| 描述 | 停车场运维人员查看停车数据统计结果 |
| 执行者 | 停车场运维人员 |
| 前置条件 | 停车场运维人员登录系统 |
| 后置条件 | 无 |
| 基本流 | 1. 停车场运维人员点击“停车数据统计” 2. 服务器返回停车数据 3. Web端展示数据统计结果” |
| 备选流 | 无 |

|  |  |
| --- | --- |
| 用例名 | 业主信息管理 |
| 描述 | 停车场运维人员对业主信息进行增删改查 |
| 执行者 | 停车场运维人员 |
| 前置条件 | 停车场运维人员登录系统 |
| 后置条件 | 无 |
| 基本流 | 1. 停车场运维人员点击“业主信息管理” 2. 服务器返回业主信息数据 3. 停车场运维人员对数据进行操作并点击“保存” 4. 服务器保存用户所做的修改 |
| 备选流 | 3a.用户未点击“保存”，直接退出系统，流程结束 |

|  |  |
| --- | --- |
| 用例名 | 用户黑名单管理 |
| 描述 | 停车场运维人员对用户黑名单进行增删改查 |
| 执行者 | 停车场运维人员 |
| 前置条件 | 停车场运维人员登录系统 |
| 后置条件 | 无 |
| 基本流 | 1. 停车场运维人员点击“用户黑名单管理” 2. 服务器返回用户黑名单数据 3. 停车场运维人员对数据进行操作并点击“保存” 4. 服务器保存用户所做的修改 |
| 备选流 | 3a.用户未点击“保存”，直接退出系统，流程结束 |

## 类图

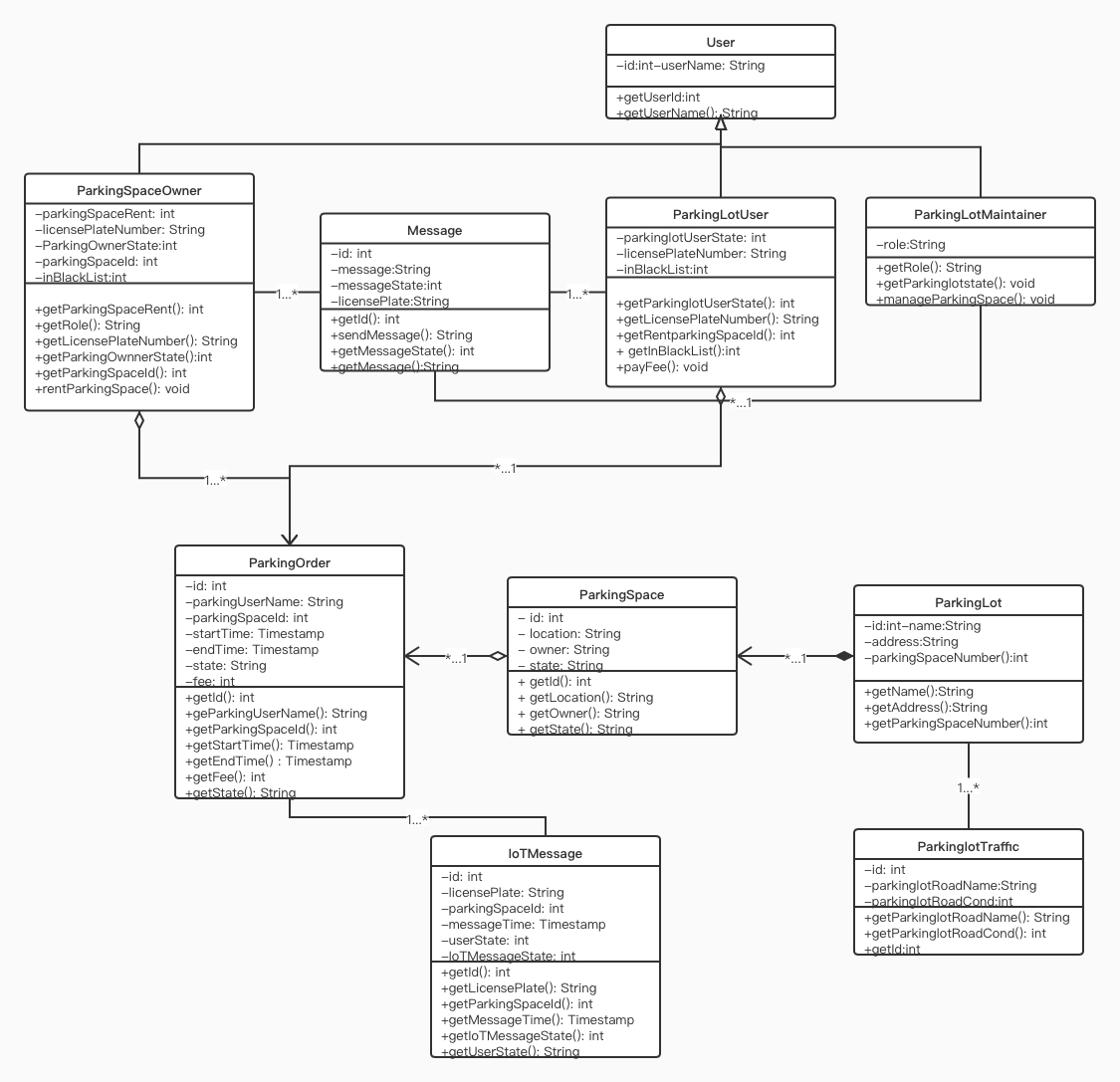


图4-1 类图

本项目的类图包括停车位业主、停车用户、停车场运维人员，这三个类都派生自用户类，但属性和方法有所不同，停车用户拥有基本的停车方法和相应的属性，停车位业主可以出租他拥有的车位，而车场运维人员可以对停车场进行管理。除此之外还有停车订单、停车位、停车位交通状况、消息、物联网系统消息和停车场类，他们分别描述了订单的属性和方法以及停车位、停车场交通状态、消息、物联网系统消息和停车场的基本属性和方法，一个停车位业主可以拥有多个停车位，一个停车位可以对应多个停车订单，一个停车场可以拥有多个停车位，每个停车用户可以拥有多个停车订单，而停车订单在同一之间只能属于一个停车用户，一个停车订单可以和多个物联网系统消息关联，一个消息可以由一个运维人员发送，一个运维人员可以发送多条消息，一个停车用户和业主可以接受多条消息。

## 数据流图

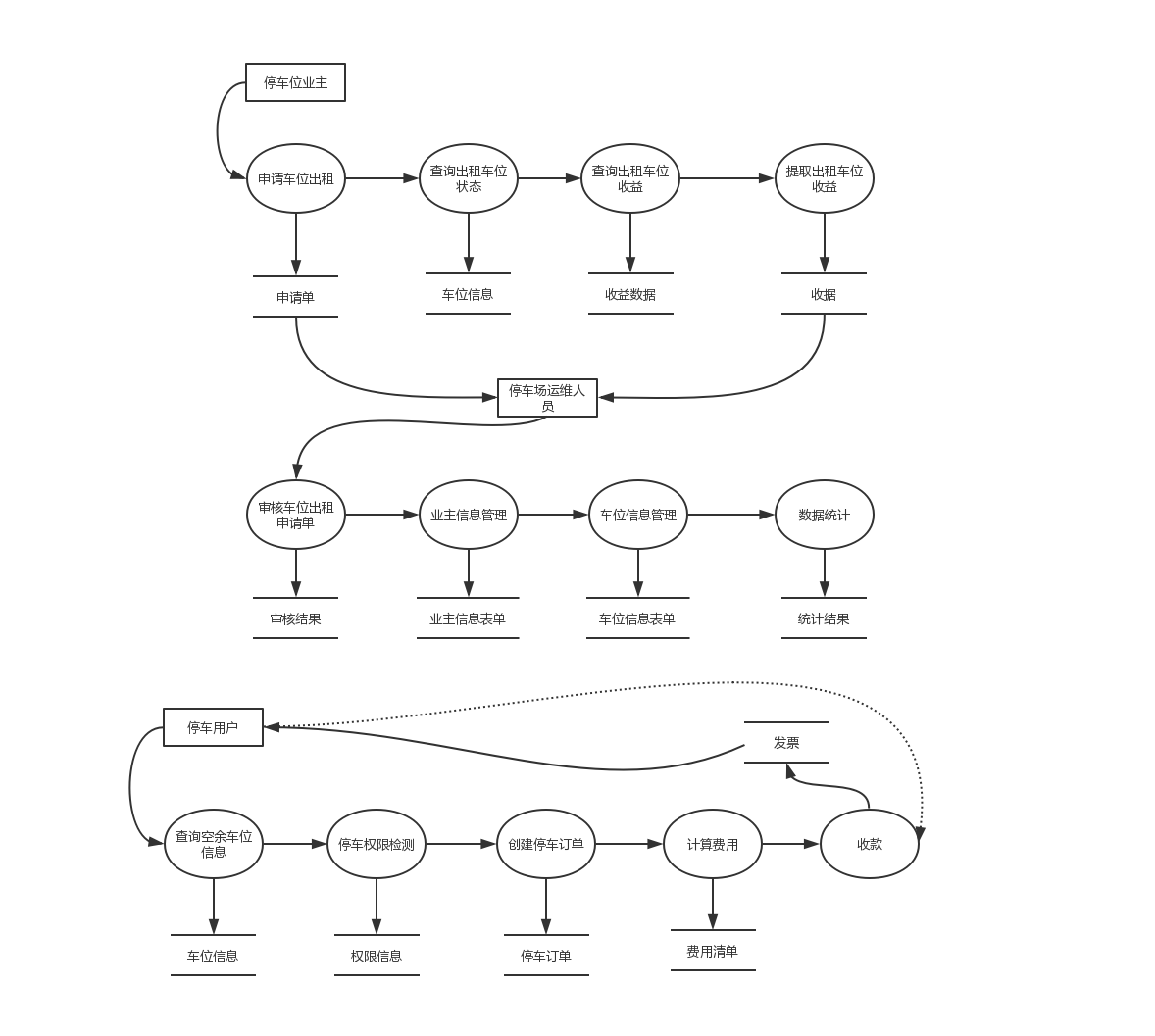


图5-1 数据流图

停车用户将用户信息提交给系统检测，检测完成后若成功则生成停车订单存入数据库中，停车为业主将车位信息提交系统进行出租审核，生成车位出租信息存入数据库。停车用户会受到计算的停车费用，停车为业主可以查询余额。停车场运维人员可以进行停车场数据的统计。

## 状态图

### 6.1 状态图1

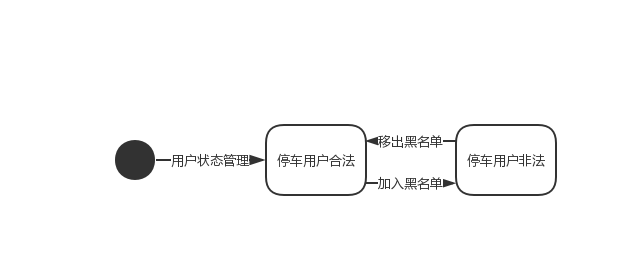


图6-1 状态图1

用户状态分为两种，初始都是合法的停车用户，运维人员可以进行操作，是否将用户加入黑名单，则这个时候用户变成非法用户，无法停车。

### 6.2 状态图2

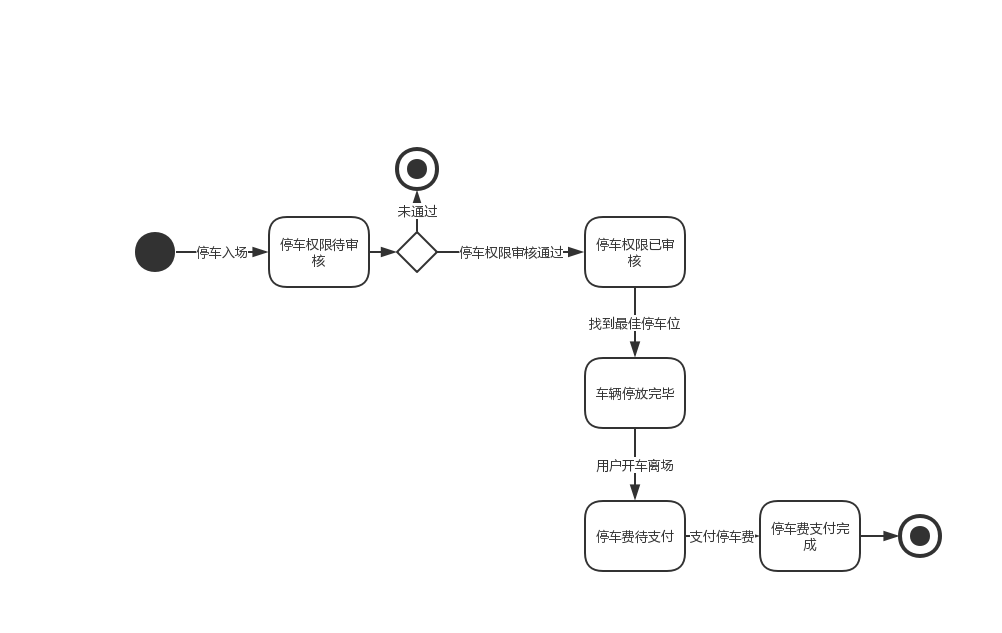


图6-2 状态图2

停车时，首先要审核停车权限，审核通过后状态发生改变，停车完毕后，需要支付停车费用，在支付停车费用后状态改变，停车费用支付完成，结束流程状态。

## 核心算法

最优路径导航即寻找从停车场入口位置到目标车位的时间最短路径。

实际生活中停车场内的车辆交通强度是实时变化，不能直接用两点间最短路径。在最短路径车辆多的情况下，路径长的路线可能耗时更短。我们使用了考虑实时交通信息的时间最短路径导航算法。

停车场路网相对较为简单,主要由环路、交叉路口及停车位等节点组成,可以抽象为一个加权有向图, 在 中,P表示停车场路网中的节点,C 表示场内有向路段,T 是权重, 其值为车辆在道路节点 i 和 j 之间的行驶时间：

式中:和 分别为节点 i 到 j 之间的距离和行驶速度。

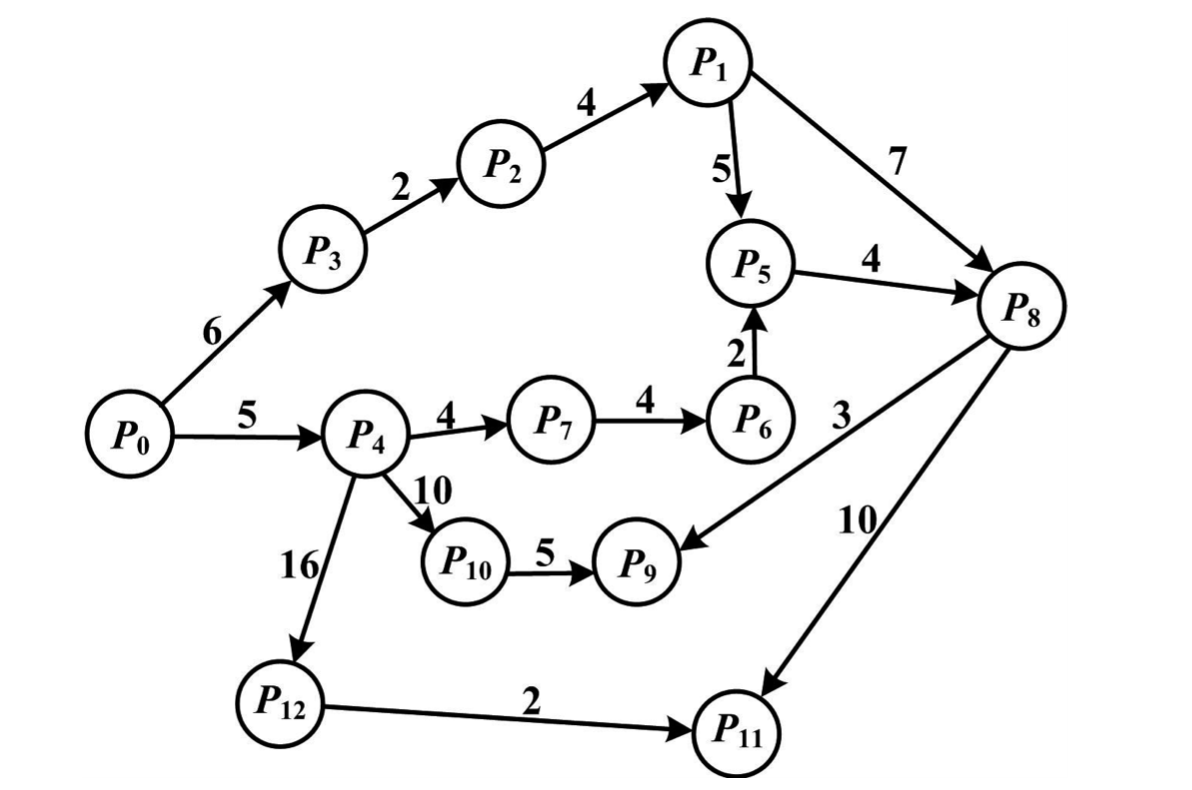


图7-1 停车场加权有向图

由公式(1)可知，图 7-1 中两个相邻节点之间的权重即车辆在该路段行驶所需要的时间。一般情况下, 道路的行驶速度 比较稳定，且与道路上的机动车数成反比。车辆数量越多，道路越拥挤，行驶速度越慢。

设定车辆在场内道路上行驶单位长度需要的时间为Δt,考虑到同一路径上多个车辆之间的相互影响, 定义延迟系数K。



其中:  。

基于时间最短路径的停车场车位引导算法，输入一个以空闲停车位为节点的邻接矩阵 ，在邻接矩阵中以停车场入口作为源顶点。用P表示所有空闲停车位节点集合，邻接矩阵 中的每一个元素表示有序节点对之间的权重。

本算法以两点之间的车辆行驶时间作为权重，不同于经典 Dijkstra 算法，该权重是随着同一时间场内环路上车辆的数目而时变的，具体变化关系如式(2)、式(3)所示。 若 P、P不相邻,则将元素 置为∞。设S为已经查找到的从出发的最短路径的节点集合，任意两节点之间的总开销就是最短路径经过的所有边的权重总和。用T表示这些最短路径的花费值，表示从源节点P出发到终点P的最短路径的开销。

算法具体步骤如下。

1 初始化最短路径集合S及其开销T，即

2 比较集合S外部各节点，选取其中最小的节点 ，则 就是目前求得的一条从 出发的最短路径的终点，并将节点加入集合S。

3 更新从到最短路径的开销值，令：

4 重复步骤2、3，直到有向图中各节点均加入集合S，即得出从到其余各节点的时间最短路径。

## Petri网

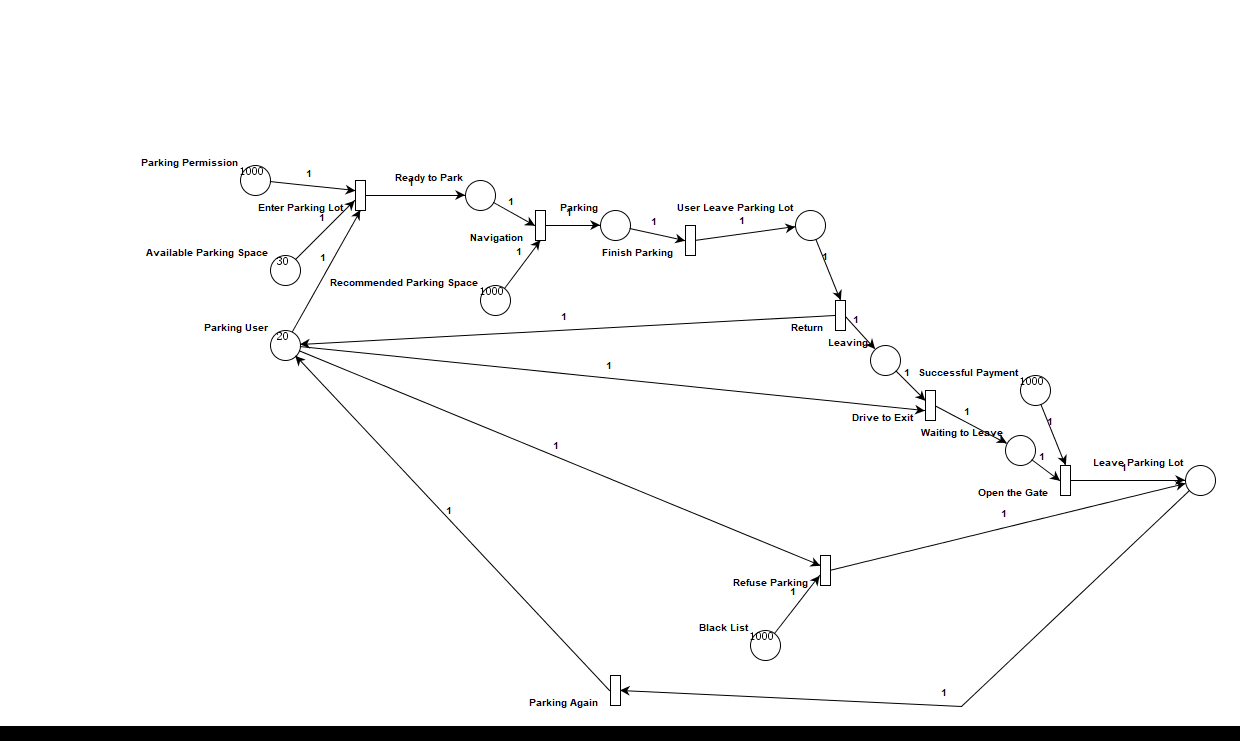


图8-1 停车流程的Petri网

PetriNet描述：初始状态时，我们会给停车用户，可用停车位，停车权限，黑名单，推荐车位和成功支付等Place赋予一些Token.当停车用户有停车权限且停车场有空余的停车位时，用户可以进入停车场，如果用户在黑名单中，则他会被拒绝进入停车场，只能离开。当用户进入停车场后，他会通过导航找到推荐车位并开始停车。停车结束后，用户会暂时离开停车场。一段时间后，用户回到停车场并准备驾车离开，在支付停车费用成功后，闸机打开，用户离开停车场。之后，用户可以再次使用停车场。

### 8.1 Petri网仿真

参数设置：我们的主要停车流程的初始条件包括停车权限、停车用户、可用停车位以及黑名单、推荐车位和成功支付。其中，停车权限可以拥有无限多的Token，主要取决于有多少停车用户，停车用户这里假设有20个，并假设可用停车位有30个，黑名单也被设为很大，考虑黑名单用户可能多次进入停车场，其余Token数设置远远大于用户数以确保流程的正常进行。速率方面，设置驶入和驶离的IoT交互事务为20，停车位10，导航位15，用户停完车离开位1，再次停车位2，体现了真实情况下的处理时间

仿真的Firings设置为100，replication设置为5。

结果：仿真结果如下图所示：

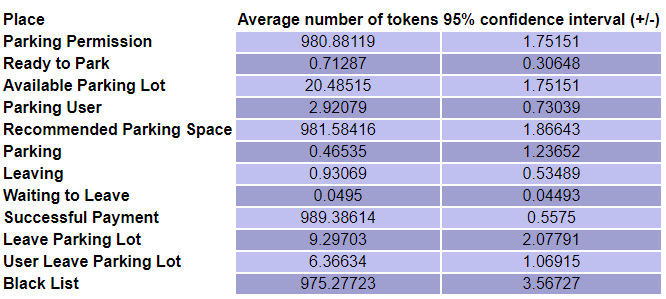


图8-2 Petri网仿真结果

### 8.3 结果分析

关键路径：如图所示，本流程的关键路径在于正常的停车的整个流程，整个关键路径的耗费时间主要跟IoT系统的响应速度，停车用户离开的时间以及寻找车位和停车的时间有关。

性能指标：本系统的关键性能指标在于服务器的响应速度，也就是在用户驶入停车场和驶离停车场时IoT系统能否及时上传数据并返回道闸的控制命令，不让用户产生堵塞，通过仿真结果可以看到，如果进入和离开的延时设置的很高的话，就会使得Token大量堆积，造成拥堵，因此这里是我们需要在设计中留意的部分。

资源能力：本系统的主要资源能力在于可用停车位的数量，一旦停车位数量变少，那么就会产生大量的拥挤和客户的流失，一旦将停车位数量减少，那么整个流程的Token数都将减少，造成停车场流量的损失。

## 分工情况

图及其余内容均为合作讨论完成。

比例分别为：

赵铭心：1/3

秦皓喆：1/3

杨震旦：1/3