# 1 业务逻辑

## 1.1 应付款计算模块



### 1.1.1 关键点

**主要分计算停车时长、计算停车费两个子模块。其中，计算停车时长要考虑折扣时间是否失效；停车费需要考虑四个计费阶梯：免费、起步、封顶和超出自然日的情况。**

### 1.1.2 伪代码

// 计算停车时长

// CurrTime为当前时间

Function calcParkedTime():

If (CurrTime - Max(dbo.payinfo.Pay\_time)) < Zk\_Time: // 是否超出折扣时间

parkedTime = Max(dbo.payinfo.Pay\_time) - InTime; // 否: 以最近一次支付时间计算

Else: // 超出折扣时间，失效

parkedTime = CurrTime - InTime

// 计算总停车款项

// 约定：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO. | 数据项 | 数据源 | 备注 | 示例 |
| 1 | **免费停车时长（分钟）** | 手动录入 | 必填,免费停车时长,freeParkMins | 120 |
| 2 | **起步收费时长（分钟）** | 手动录入 | 必填,规则内一口价,baseFareMins | 180 |
| 3 | **起步收费金额（角）** | 手动录入 | 必填,规则内一口价,baseFare | 100 |
| 4 | **全天最高限额（角）** | 手动录入 | 必填,dayLimitFare | 200 |
| 5 | **超出起步价后计费规则，每（分钟）** | 手动录入 | 必填,计费单位时长,billingUnitMins,不足一计费时间单位按照一时间单位计算 | 10 |
| 6 | **超出起步价后计费规则，收费（角）** | 手动录入 | 必填,每计费单位时长内收取费用,farePerUnit | 10 |

Function calcTotalAmount():

If parkedTime < freeParkMins : then return 0 // 免费

Elseif parkedTime < baseFareMins : then return baseFare // 起步收费

Elseif parkedTime < 24 \* 60 : then // 一个自然日内的计费

noCeilFare = baseFare + ceil((parkedTime - baseFareMins) / billingUnitMins) \* farePerUnit // 不封顶价格

return noCeilFare < dayLimitFare ? noCeilFare : dayLimitFare; // 最终价格

Else : then

ceilFare = parkedTime / ( 24 \* 60 ) \* dayLimitFare

noCeilFare = parkedTime % ( 24 \* 60 ) / billingUnitMins \* farePerUnit

return ceilFare + (noCeilFare < dayLimitFare ? noCeilFare : dayLimitFare);

// 本次停车已付款项

Function alreadyPaid():

SELECT SUM(Amount) INTO Amount From dbo.payinfo WHERE Plate = :Plate: AND InTime = :InTime: AND OutTime IS NULL;

Return Amount

// 应付款项

Function shouldPay():

Return totalAmount - alreadyPaid

## 1.2 车牌查询菜单



### 1.2.1 关键点

**判断是否有应付金额，再确定是否显示支付通道。现在有的停车场存在微信支付0元的情况。**

### 1.2.2 查询语句

SELECT \*

FROM (SELECT t.\*

FROM dbo.stopedinfo t

WHERE t.plate = :plate: AND t.out\_time = NULL

ORDER BY t.in\_time ASC)

WHERE ROWNUM < 2;

## 1.3 停车缴费



## 1.4 操作员登录



## 1.5 收费配置

### 1.5.1 配置数据的自动加载



### 1.5.2 修改配置数据的即刻生效



关键点：因配置数据量较少，Web App层使用ConcurrentHashMap保存配置。

# 2 数据模型

## 2.1 概念模型



## 2.2 物理模型



# 3 待确认的问题

1. 异构场景（约束项）：当存在两条及以上入场记录（无出场时间），应选择最早入场记录还是选择最近入场记录？
2. 性能考虑：dbo.stopedinfo是否有分区字段，是否能缩小查询范围，数据生命周期是怎样的？
3. 操作员账号、密码在车安库建还是在我们这边建？是否要增加账号维护的功能，如增删改？
4. 业务规则：跨自然日计费，对一个自然日的理解，应以固定时间点计算（如0:00~23:59）还是以入场时间为起点？是否不足一个自然日的部分不再享受起步收费，因第一个自然日已享受起步收费。
5. 业务规则：对于不同种类号牌的支持，如新能源车需要增加一位，武警号牌车允许自行录入车牌（无空格）；