基于Java的3D智能停车系统之停车管理子系统的设计与实现

# 背景及现状分析

## 研究背景

随着互联网的高速发展，从2G到3G，再到如今5G即将开放，多样的信息技术已经进入各个行业中，在4G时代，人们已经可以通过互联网进行购物，买票，吃饭。在5G的物联网时代，万物互联，各种基础设施与互联网连接已成为可行的现实，极大程度上改变了人们的生活。如今5G时代汽车的无人驾驶技术蓬勃进步，同样的配套设施——车库也应当与时俱进，加入信息技术，使这种基础设施散发第二春。

## 现状分析

如今的车库大部分还是2D的平面车库，用户需要将车从入口开入，寻找车位存车，停车之后走出车库。在取车的时候还需要步行进入车库，开走车辆，在出口缴费才能开走车辆，其中有很多的步骤是可以省去的，由于人车混杂，容易造成安全事故，且车库空间封闭，发生灾害很难逃生，考虑到安全性，3D车库由于没有人在车库，全靠机械机构进行运行，安全可靠。

在国外存在3D车库，但是由于国外互联网的大环境不如国内，尤其是互联网付费系统不如国内成熟，而且国外大城市的车流量不足，没有够大的停车需求，从而难以构建出使用手机进行存取车的3D立体车库。

手机是如今个人必不可少的工具，人们的衣食住行各个离不开手机，手机的便捷性使得手机可以作为个人移动终端进行人们的日常活动。

如今数据分析技术以成为主流，车库的存车、取车情况日积月累也是一种大数据，如何合理利用这些数据，分析这些数据，从数据中得到关联，从数据中“掘金”，从得出的结论中优化系统也是一个大难题。

3D立体车库如今存在可行的技术实现，但是比较缺少与信息技术连接的“3D立体智能车库”，本设计将提供一种可供立体车库进行运行，管理，监控，数据分析的方案。

# 需求分析

## 引言

## 命名标准和规范

## 系统功能性需求

## 非功能性需求

### 可靠性

### 可用性

### 性能

### 可支持性

### 安全性

# 模拟车库设计

# 系统设计

# 数据库设计

## CAR

汽车表，用于存放用户汽车信息。

### ID

VARCHAR类型，使用系统中的方法自动生成16位id，主键。

### USERID

VARCHAR类型,对应用户表的ID字段。

### CARCARD

VARCHAR类型，汽车的车牌号，不可重复。

### CARNAME

VARCHAR类型，加密，汽车的名称。

### CARINFO

VARCHAR类型，加密，汽车的信息。

## CONTRAST

对照表，一个ID对应一个真实姓名。

### ID

VARCHAR类型，使用系统中的方法自动生成16位id，主键。

### REALNAME

VARCHAR类型，加密，这条数据所代表的真实名称

## EMAILACTIVE

电子邮箱激活表。

### ID

VARCHAR类型，使用系统中的方法自动生成16位id，主键。

### USERID

VARCHAR类型，USER表中的ID字段

## PARKING

停车表，用于存放停车信息。

### ID

VARCHAR类型，使用系统中的方法自动生成16位id，主键。

### CARID

VARCHAR类型，对应汽车表中的id字段。

### INSPACE

VARCHAR类型，车辆入库的地点，一般为入口，VIP存车为缓冲区。

### NOWSAPCE

VARCHAR类型，现在车辆所在的位置。

### SAVESPACE

VARCHAR类型，车辆的存车地点。

### OUTSPACE

VARCHAR类型，车辆的取车地点，一般为出口或者缓冲区。

### VIPSENDTIME

DATETIME类型，用户预约停车时的时间。

### VIPAPPPARKINGTIME

DATETIME类型，用户预约停车的时间。

### INPLACETIME

DATETIME类型，车辆入库的时间。

### INTIME

DATETIME类型，车辆停车的时间。

### VIPGETTIME

DATETIME类型，用户预约取车时的时间。

### VIPAPGETTIME

DATETIME类型，用户预约取车时间。

### OUTTIME

DATETIME类型，取车时间。

### OUTPLACETIME

DATETIME类型，车辆离开车库的时间。

### GETTIME

DATETIME类型，用户取走车辆的时间。

### NATURE

VARCHAR类型，性质（等待用户存车中，开往停车位，停车中，取车中，等待用户取车中）

### WAY

TEXT类型，车辆经过的路径。

## PARKINGSAVE

停车保存表，这个表用于存储历史停车信息。

### ID

VARCHAR类型，使用系统中的方法自动生成16位id，主键。

### CARID

VARCHAR类型，对应汽车表中的id字段。

### INSPACE

VARCHAR类型，车辆入库的地点，一般为入口，VIP存车为缓冲区。

### SAVESPACE

VARCHAR类型，车辆的存车地点。

### OUTSPACE

VARCHAR类型，车辆的取车地点，一般为出口或者缓冲区。

### VIPSENDTIME

DATETIME类型，用户预约停车时的时间。

### VIPAPPPARKINGTIME

DATETIME类型，用户预约停车的时间。

### INPLACETIME

DATETIME类型，车辆入库的时间。

### INTIME

DATETIME类型，车辆停车的时间。

### VIPGETTIME

DATETIME类型，用户预约取车时的时间。

### VIPAPGETTIME

DATETIME类型，用户预约取车时间。

### OUTTIME

DATETIME类型，取车时间。

### OUTPLACETIME

DATETIME类型，车辆离开车库的时间。

### GETTIME

DATETIME类型，用户取走车辆的时间。

### NATURE

VARCHAR类型，性质（等待用户存车中，开往停车位，停车中，取车中，等待用户取车中）

### WAY

TEXT类型，车辆经过的路径。

## PHONEACTIVE

手机激活表

### ID

VARCHAR类型，使用系统中的方法自动生成16位id，主键。

### USERID

VARCHAR类型，保存用户的ID。

### CODE

验证码,加密，保存用户的手机验证码信息。

## SPACE

空间表，保存车库模型

### ID

VARCHAR类型，使用系统中的方法自动生成16位id，主键。

### X

INT类型，车库的位置坐标X轴。

### Y

INT类型，车库的位置坐标Y轴。

### Z

INT类型，车库的位置坐标Z轴。

### NATURE

VARCHAR类型，空间的性质（入口/出口，通道-可通过/占用，车库-空置/占用，缓冲区-空置/占用，已激活/未激活/会员，待用户存车中，开往停车位，停车中，取车中，等待用户取车中）

### CARID

VARCHAR类型，汽车地id。

## USER

用户表，保存用户的信息。

### ID

VARCHAR类型，使用系统中的方法自动生成16位id，主键。

### NICKNAME

VARCHAR类型，加密，用户名。

### PASSWORD

VARCHAR类型，加密，密码。

### REALNAME

VARCHAR类型，加密，真实姓名。

### IDCARD

VARCHAR类型，加密，身份证号。

### PHONE

VARCHAR类型，加密，手机号。

### PHONENATURE

VARCHAR类型，是否激活手机。

### EMAIL

VARCHAR类型，邮箱号

### EMAILNATURE

VARCHAR类型，是否激活邮箱。

### NATURE

VARCHAR类型性质，0未激活，1激活，2会员。

## VIP

VIP表

### ID

VARCHAR类型，主键

### USERID

VARCHAR类型，对应USER表中主键

### ENDDATE

ENDDATE类型，会员截止日期

# 算法设计

## 获得最佳车位算法

### 分析

空间表（SPACE）的三维坐标（X，Y，Z）代表停车位在车库的具体位置，最佳位置即从入口到停车位再到出口路程最短的车位。

同一列中，入口到车位的距离与车位到出口的距离相加是一样的，保证减少阻塞，所以最佳车位为最右边，即X越大越好。

同一层中，前排车位的从入口到车位的距离与从车位到出口的距离相加最小，最佳车位为最靠前的一排，即Y越小越好。

同一车库中，最顶层的车位的从入口到车位的距离与从车位到出口的距离相加最小，最佳车库曾为最上层，即X越小越好。

### 结果

每一排的车位数量为MAX\_X，每一层有MAX\_Y排，停车位的坐标为（X，Y，Z）所以每一个最佳停车位的权重为：

X + Y \* MAX\_X + z \* MAX\_X \* MAX\_Y

权重越小，停车位置越好。

## 存车算法

### 分析

共有5种类型的通道：

入口所在的层数变化通道，只能向下一层移动。

出口所在的层数变化通道，只能向上一层移动。

每一层的左侧通道，只能向下移动。

每一层的右侧通道，只能向上移动。

每一层每一排的通道，只能向右移动。

故可以将每个车辆的停车路线进行原子化，对每一个车辆的下一步进行计算，获得一下情况。

车辆在入口位置。

车辆在左侧通道。

车辆在横向通道。

因为每一层的每第一排只能向上移动，所以所有的非第一排需要向下移动进入停车位。

### 结果

车辆所在的位置的三维坐标为（X，Y，Z），停车位位置为（FX，FY，FZ）。

若在入口位置，所在位置与停车位所在位置的层数不一，则向下一层移动。

若在左侧通道，所在位置的右侧不是横向通道，则向下移动。

若在左侧通道，所在位置的右侧是横向通道，但是停车位所在的排与现在位置所在的排相减的绝对值不为1（这个横向通道无法进入车位），则向下移动。

若在左侧通道，所在位置的右侧是横向通道，所在位置的右侧是横向通道，停车位所在的排与现在位置所在的排相减的绝对值为1（这个横向通道可以进入车位），则向右移动。

若在横向通道，所在位置与停车位的列数不一致，则向右移动。

若在横向通道，所在位置与停车位的列数一致，停车位置在所在位置上方，则向上移动。

若在横向通道，所在位置与停车位的列数一致，停车位置在所在位置下方，则向下移动。

所以得到如下算法：

如果是不同层的，往下一层挪动。

如果是左侧通道，且停车位置与现在车辆所在位置的排绝对值为不为1，则向下移动。

如果停车位置与现在车辆所在位置的排的绝对值为1，所在位置与停车位置的列数不同，则向右移动。

如果所在位置与停车位置的所在列数相同，但不同排，则上下移动。

## 取车算法

### 分析

经过分析，共有5种类型的通道：

入口所在的层数变化通道，只能向下一层移动。

出口所在的层数变化通道，只能向上一层移动。

每一层的左侧通道，只能向下移动。

每一层的右侧通道，只能向上移动。

每一层每一排的通道，只能向右移动。

故可以将每个车辆的取车路线进行原子化，对每一个车辆的下一步进行计算，获得一下情况。

### 结果

车辆在左侧通道并且可以往右移动，则向右移动

车辆在左侧通道但是不能往右移动，只能向下移动

车辆在车位上，且在第一排，往下挪动。

车辆在车位上，且不在第一排，可以向上移动。

车辆不在车位上，且不在最右侧通道，只能向右移动。

车辆不在车位上，且不在同一排，所以只能在右侧通道,只能往前挪动

车辆只能是楼层不同，则向上移动。

# 功能设计

## 用户信息模块

### 注册

#### Android

获得手机号后进行加密后发送到后台。

#### 后台

接收用户手机号码（加密）后进行解密。

从数据库中匹配是否有手机是重复的

如果是重复的，返回加密后的FALSE。

如果不是重复的，进行下面步骤：

设置初始密码123456，然后进行加密。

设置用户昵称为新用户+手机号，然后进行加密。

设置用户性质NATURE为未激活，不进行加密。

设置PHONENATURE为未激活，不进行加密。

设置EMAILNATURE为未激活，不进行加密。

将加密后的手机号，密码，昵称，性质，PHONENATURE，EMAILNATURE添加进表USER。

通过对照表获得PHONENATURE和EMAILNATURE字段的对照名称，将其与PHONENATURE和EMAILNATURE进行替换。

将USER信息（被加密）返回前台。

#### Android

获得返回信息，将返回信息进行解密。

若获得信息FALSE，则提示用户已被注册。

若获得信息为USER数据，则跳转到用户主页。

### 登陆

#### Android

获得用户的手机号，密码信息，在前端进行加密，发送到后台。

若获得的数据为NULL则提示用户名或密码错误。

若获得的数据不为NULL则跳转到用户主页，将数据解密在页面展示。

#### 后台

获得加密后的手机号，密码信息，在数据库进行匹配。

若匹配不上，则返回NULL。

若匹配上，则根据被加密的手机号获得用户信息。

对用户信息的REALNAME，IDCARD，PHONENATURE，EMAILNATURE进行解密。

如果PHONENATURE为1且EMAILNATURE为1且REALNAME不为空且IDCARD不为空，则NATURE为1。其他情况为0。

若NATURE为1且VIP表中的USERID存在且其中一条数据的ENDDATE大于现在的时间，则NATURE为2。

将USER更新到表中。

根据USER表的ID字段获得CAR表的信息。

对USER，CAR信息进行加密。

返回加密后的USER信息，CAR信息。

### 邮箱激活

#### Android

获得邮箱信息和用户手机号进行加密发送到后台。

若获得返回的数据为NULL，提示邮箱已存在。

若返回的数据不为NULL，解密数据。

若返回的数据为FALSE，提示邮箱发送失败，请检查地址。

若返回的数据为TRUE，则提示邮件发送成功。跳转到主页。

用户登陆邮箱，点击邮件提供的链接。

若返回的信息为FALSE，则提示激活失败，换一个邮箱。

若返回信息为TRUE，则提示激活成功，跳转到登陆页面。

#### 后台

获得邮箱进行数据库匹配，若有重复的，则返回邮箱已存在。

若没有重复的，则新建邮箱激活类EMAILACTIVE，添加静态方法生成ID。

对用户手机号进行匹配，获得用户ID，添加进EMAILACTIVE的USERID。

将USER信息的EMAIL添加进EMAIL。

向数据库添加EMAILACTIVE。

根据用户激活类的ID编写邮件进行发送。

若发送成功，返回加密后的TRUE。

若发送失败，返回加密后的FALSE。

后台接收到激活信息，对数据库进行匹配，若匹配失败，则返回false。

若匹配成功，删除EMAILACTIVE表中的这个用户所有的EMAILACTIVE信息。

根据手机号获得USER，添加EMAIL，将EMAILACTIVE字段改为加密后的1。

更新USER。

返回TRUE。

### 手机激活

#### Android

发送用户手机号进行加密到后台。

如果返回FALSE则提示发送失败。

入伙返回TRUE则提示验证码发送成功。

发送用户验证码和手机号进行加密到后台。

如果返回FALSE，则提示验证码错误。

如果返回TRUE，则跳转到主页。

#### 后台

获得用户手机号，根据手机号获得用户实体类。

新建手机激活类PHONEACTIVE

添加ID，用户ID，生成四位验证码。

PHONEACTIVE添加到数据库。

发送验证码到手机。

若发送失败，则返回FALSE。

若发送成功，则返回TRUE。

接收PHONEACTIVE进行数据匹配。

若匹配失败，则返回FALSE。

若匹配成功，则用手机号获得用户USER。

将USER的PHONENATURE字段改为1。

如果PHONENATURE为1且EMAILNATURE为1且REALNAME不为空且IDCARD不为空，则NATURE为1。其他情况为0。

若NATURE为1且VIP表中的USERID存在且其中一条数据的ENDDATE大于现在的时间，则NATURE为2。

将USER更新到表中。

返回TRUE。

### 实名认证

#### Android

发送用户姓名，身份证号进行加密发送到后台。

若后台返回SAME则提示用户已存在。

若后台返回FALSE则提示实名认证失败。

若后台返回TRUE则提示实名认证成功。

跳转到主页。

#### 后台

接收到身份证号与真实姓名进行解密。

在数据库中进行匹配。

若匹配不为1，则返回SAME。

将身份证号与真实姓名进行实名认证。

若实名认证失败，则返回FALSE。

若实名认证成功，则通过PHONE获得用户。

将身份证号与真实姓名进行加密，放到用户信息中。

如果PHONENATURE为1且EMAILNATURE为1且REALNAME不为空且IDCARD不为空，则NATURE为1。其他情况为0。

若NATURE为1且VIP表中的USERID存在且其中一条数据的ENDDATE大于现在的时间，则NATURE为2。

将USER更新到表中。

返回TRUE。

### 添加车辆

#### Android

前端发送车牌号，汽车名，汽车备注进行加密发送到后台。

若返回SAME则提示该车已被添加。

若返回为FALSE则提示添加失败。

若返回TRUE则提示添加成功，挑战到主页。

#### 后台

获得加密后的车牌号，汽车名，汽车备注并进行解密。

对车牌号进行数据库匹配，若数量大于0，则返回SAME。

对车牌号进行认证，若不匹配则返回FALSE。

若匹配则根据用户手机号新建CAR，CAR加入汽车信息，加入数据库。

返回TRUE。

### 修改车辆信息

#### Android

前端发送新旧车牌号，汽车名，汽车备注进行加密发送到后台。

若返回SAME则提示该车已被存在。

若返回为FALSE则提示修改失败。

若返回TRUE则提示修改成功，挑战到主页。

#### 后台

获得加密后的新旧车牌号，汽车名，汽车备注并进行解密。

对新车牌号进行数据库匹配，若数量大于0，则返回SAME。

对新车牌号进行认证，若不匹配则返回FALSE。

若匹配则根据旧车牌号获得CAR，CAR加入汽车信息，进行加密，更新数据库。

返回TRUE。

## 停车取车模块

### 用户停车

#### Android

用户查看车库是否存在空车位。

用户选择车辆，点击停车，扫入口二维码。

若是返回FALSE表示停车失败。

若是返回TRUE表示停车成功。

#### 后台

接收到二维码的数据，进行解密，解密后进行验证。

若验证失败，则返回FALSE。

从数据中得到汽车编号CARID

使用获得最佳位置的空车位算法获得空车位SPACE

SPACE修改NATURE为“车位-占用”。

SPACE修改CARID为获得的汽车数据的编号。

新建PARKING对象，ID设置为自生成，INTIME设置为当前时间，SAVEPLACE设置为空车位SPACE的编号，NOWSPACE设置为入口编号，NATURE设置为“开往停车位”。

将PARKING添加进静态变量。

更新静态变量SPACES与PARKINGS。

返回TRUE。

### VIP用户预约停车

#### Android

客户预约：

用户查看车库是否存在空车位与空缓冲区。

用户选择车辆，点击预约停车，选择停车时间，点击确定。从而获得停车地点。

客户停车：

用户将车停入停车地点，选择车辆，点击停车，扫车位的二维码。

#### 后台

客户预约：

接收数据后解密，获得存车时间，车辆信息。

使用获得最佳位置的空车位算法获得空车位SPACE，NATURE为“车位-占用”，CARID为获得的汽车数据的编号。

使用获得缓冲区算法获得缓冲区SPACEFORCASH，NATURE修改为“缓冲区-占用”， CARID修改为汽车编号。

新建PAKRING保存停车信息，ID使用自生成算法获得， VIPSENDTIME为当前时间，VIPAPPARKINGTIME为获得的时间，NATURE为“等待用户停车“，SAVEPLACE为获得的车位ID。

在缓冲区修改PAKINGS与SPACES。

客户停车：

接收到二维码信息进行验证，对接收到的车辆信息进行匹配。

通过二维码的缓冲区信息获得缓冲区车位SPACE

SPACE设置NATURE为“缓冲区-空置”，NATURE修改为null。

通过用户车辆信息获得停车信息PAKRING，将INTIME修改为当前时间，NATURE修改为“开往空车位”，NOWSPACE修改为入口ID。

在缓冲区修改PARKINGS与SPACES。

### 用户取车

#### Android

用户获得车辆：

用户选择车辆，点击取车，扫出口二维码，点击确认。

用户开走车辆：

用户选择车辆，点击确认取车，扫出口二维码，点击确认。

#### 后台

用户获得车辆：

解密数据，将数据进行验证。

若验证失败则返回FALSE。

获得加密后的信息进行解密，根据车辆信息在静态变量中获得停车信息(PARKING)，将他的的取车时间(OUTTIME)设置为当前时间，性质(NATRUE)为“取车中”。

更新静态变量。

用户开走车辆：

解密数据，将数据进行验证。

若验证失败则返回FALSE。

通过车辆信息获得停车信息，将停车信息中的TAKEOUTTIME设置为当前时间。

新建PARKINGSAVE对象，复制PAKRING信息，进行存储，静态变量中删除PAKRING信息。

### VIP用户预约取车

#### Android

预约取车：

用户将取车时间，车辆信息加密后传向后台，得到取车地点。

若无缓冲区则提示无法预约取车。

取车：

用户到达缓冲区指定位置，选择车辆，点击预约取车，扫码。

#### 后台

预约取车：

后台接收到车辆信息，取车时间。

根据获得缓冲区算法获得缓冲区。将其性质设置为“缓冲区-占用“ ，汽车编号设置为车辆信息的汽车编号。

汽车信息的性质设置为“VIP取车中”，预约取车时间（VIPAPPGETTIME）为获得的取车时间，VIPGETTIME为当前时间。

更新空间链表和停车信息链表。

返回缓冲区信息。

取车：

验证二维码。

获得停车信息，将取走时间设置为当前时间。

修改缓冲区的性质为“缓冲区-空置”，汽车编号为空。

## 车库运行

### 停车

从静态变量中获得所有正在开往车位的停车信息（PAKRING）链表，即性质(NATUE)为“开往停车位”，并对他们的入库时间(INTIME)进行从早到晚的排序，并逐一对他们进行如下处理：

根据停车信息获得现在所在的位置（NOWSPACE），停车位（FETURESPACE）和根据存车算法获得的下一步位置（NEXTSPACE）。

如果下一步位置的汽车编号为空，则将停车信息的路径增加“NULL”，更新停车信息链表，跳过本次循环。

将现在位置的车辆编号设置为空，性质为“通道-可通过”，下一步位置的车辆编号设置为汽车编号，性质为“通道-占用”。

将停车信息的路径增加下一步位置的信息，将现在位置修改为下一步位置。

如果现在位置是入口，则修改现在的位置性质为“入口”。

如果下一步的位置是车位，则修改停车信息的性质为“存车中”，下一步性质为“车位-占用”，停车时间(SAVEINPLACETIME)为当前时间。

更新静态变量的PARKING链表，SPACES表。

### VIP预约停车

从静态变量中获得VIP预约停车且预约时间大于当前时间的PARKING链表，即NATURE是“等待用户存车中”和VIPAPPTIME这两项。

对于它们每一个进行如下操作：

PARKING的汽车编号获得停车位置与缓冲区位置。

将这个PAKRING在静态变量中删除。

将停车位置的汽车编号设置为空，性质修改为“车位-空置”。

将缓冲区的汽车编号设置为空，性质修改为“车位-空置”。

在静态变量中更新这两项。

### 取车

从静态变量中获得所有取车的链表，按取车时间早晚进行排序。

对车辆信息逐一进行一下处理：

获得下一步位置。

判断下一步位置能否通过，如果不能通过，则对停车信息的路径添加NULL。

如果可以通过，则更新经过的路径。

现在的位置设置汽车编号为NULL，下一步位置设置汽车编号为NULL。

汽车信息修改现在位置字段为现在位置。

如果到达出口，则把性质改为“等待客户取车”，离开车库时间为当前时间，现在位置的性质改为“通道-可通过”，下一个位置的性质改为“出口”，汽车编号为空。

如果现在位置是车库，则现在位置的性质改为“车位-空置”，汽车编号为空，下一个位置的性质改为“通道-占用”。

其他情况，现在位置改为“通道-可通过”，下一个位置改为“通道-占用”

如果停车信息的VIP预约取车时间不为空且现在位置是出口，则下一个位置为缓冲区，汽车信息修改性质为“缓冲区等待取车”。

## 大数据展示监控管理模块

### 监控

视频流传输到前端页面。

### 统计

通过java8的stream方法获得数据通过echart传到前台。

### 管理

单表维护。

# 交互设计

## Android端（普通用户使用）

### 注册

点击应用，点击新用户注册，输入手机号，点击注册。

若注册成功，提示注册成功密码123456，跳转到登陆界面。

若注册失败，提示注册失败，手机号已被使用。

### 登陆

点击app输入账号密码。

若登陆成功，跳转到用户主页。

若登陆失败，提示账号密码错误。

### 激活

#### 邮箱激活

登陆后，点击激活邮箱，跳转到激活邮箱界面，输入邮箱，点击确定。

若发送失败，提示邮箱不存在。

若邮箱已存在，提示邮箱已存在。

若激活成功，返回主页。

收到邮件后点击链接

若激活成功，提示激活成功。

若激活失败，提示此邮件已过期。

#### 手机激活

登陆后点击激活手机，跳转到激活手机界面，点击获得验证码，获得后输入验证码。

若激活成功，则提示激活成功，返回用户主页。

若验证码不存在，提示验证码输入错误。

若验证码过期，提示验证码过期。

#### 实名认证

登陆后点击实名认证，跳转到实名认证界面，输入姓名和身份证，点击确认。

若激活成功，提示激活成功，跳转到用户主页。

若格式不符合，提示格式不符合，请重新输入。

### 添加车辆

点击添加车辆，输入车辆的车牌号，车名，备注信息，点击确定。

若添加成功，提示添加成功，跳转到停车页面。

若车牌号不存在，提示车牌号不存在。

若车辆已被注册，提示车辆已被注册。

### 修改信息

#### 修改手机

点击修改手机，进入修改手机界面，输入旧手机号，新手机号，点击确定。

若旧手机号正确，修改成功。

若就手机号不正确，提示修改失败。

#### 修改邮箱

点击修改邮箱，进入修改邮箱界面，输入旧邮箱，新邮箱，点击确定。

若旧邮箱正确，修改成功。

若旧邮箱不正确，提示修改失败。

#### 修改密码

点击修改密码，进入修改密码界面，输入旧密码，新密码，点击确定。

若旧密码正确，提示修改成功，返回登陆页面。

若旧密码错误，提示修改失败。

#### 修改车辆信息

点击车辆管理，选择车辆进行修改。

若车辆无记录，且车牌号符合，修改成功。跳转到车辆管理。

若不符合，提示修改失败。

### 成为VIP

点击成为vip，点击确定。支付宝支付。

### 停车

#### 正常停车

用户将车辆停入入口，选择车辆，点击停车，扫入口二维码，点击确定。

#### VIP预约停车

用户选择车辆，点击预约停车，选择停车时间，点击确定。

用户获得停车位置。

用户在指定时间之内把车停入位置。

选择车辆，点击停车，扫停车位的二维码。

### 取车

#### 正常取车

用户到达出口，选择车辆，点击取车，扫出口二维码。

#### VIP预约取车

用户选择车辆，点击预约取车，选择取车时间。

用户获得取车地点，

用户到达取车地点，选择车辆，点击取车，扫取车地点二维码。

## 大数据展示管理监控

### 监控

选择第几层和那个位置的摄像头，开始监控。

### 指标分析

点击指标分析，查看想要的指标。

### 管理维护

点击维护，选择需要维护的信息。

# 参考文献

# 附录

# 致谢