

基于 Android 的车辆远程控制 APP^①



白云伟, 李丽萍

(上海第二工业大学 计算机与信息工程学院, 上海 201209)

通讯作者: 白云伟, E-mail: baiyw@sspu.edu.cn

摘要: 针对智能驾驶的需求, 设计和开发出了一款远程控制车辆的 APP. 该 APP 在设计实现时, 采用 XMPP 的通信协议进行手机端和车载系统的通信, 用页面图标形式呈现车辆的胎温胎压、车内温度、续航里程等状况, 如有故障会亮红报警, 嵌入高德地图 API 提供远程车辆定位和导航到车位置的功能, 并具有远程车辆召唤, 车外自动泊车、远程自动开车门等功能. 该 APP 已与实车进行绑定测试, 结果表明该 APP 稳定可用.

关键词: 远程控制; 手机终端; 通信; Android APP

引用格式: 白云伟, 李丽萍. 基于 Android 的车辆远程控制 APP. 计算机系统应用, 2020, 29(2): 112-117. <http://www.c-s-a.org.cn/1003-3254/7207.html>

Vehicle Remote Control APP Based on Android

BAI Yun-Wei, LI Li-Ping

(School of Computer and Information Engineering, Shanghai Polytechnic University, Shanghai 201209, China)

Abstract: Based on the requirement of intelligent driving, an APP of remote vehicle control is designed and developed. When designing and developing this APP, we use the XMPP communication protocol for mobile and vehicle communication, and use icons to present vehicle tire pressure, tire temperature, temperature inside the car, endurance mileage, and other conditions. If the car falls in fault, the alarms arise with bright red on the page. Amap API is also embedded to provide the function of the remote vehicle location and navigating to the car position. It also has the feature of remote vehicle calling, automatic car parking, and remote automatic door opening. The APP has been tested by binding with real cars, and the results show that the APP is stable and available.

Key words: remote control; mobile terminal; communication; Android APP

汽车行业飞速发展, 生活越来越便利, 用户对汽车服务的要求越来越高, 近几年随着新能源车步入市场, 新能源车系列也受到越来越多人的青睐. 物联网时代, 本着万物互联的畅想, 手机和车机互联也随之而生, 能够运用手机 APP 来远程控制车辆成为用户的迫切需求.

远程控制 APP 利用实时通信技术, 以页面图标呈现的方式供用户实时查看车辆状态参数, 并且用户可以根据需要进行远程开关空调, 远程开关车门, 远程定位车辆, 个性化设置, 以及远程召唤车辆和车外自动泊车等操作.

1 APP 开发环境及应用技术

1.1 开发环境

远程控制 APP 用 Android Studio 3.0 作为开发平台, 服务器采用 XMPP 通信协议实时通信和 MySQL 数据库进行数据操作.

Android Studio 是 Google 推出的一个 Android 集成开发工具, 基于 IntelliJ IDEA, 类似 Eclipse ADT, Android Studio 提供了集成的 Android 开发工具用于开发和调试. 开发者可以在编写程序的同时看到自己的应用在不同尺寸屏幕中的样子. Android 是一种基于

① 收稿时间: 2019-05-21; 修改时间: 2019-06-21; 采用时间: 2019-07-01; csa 在线出版时间: 2020-01-16

Linux 的自由及开放源代码的操作系统, 主要使用于移动设备, 如智能手机和平板电脑. Android 操作系统有 4 层架构, 从高层到低层分别是应用程序层、应用程序框架层、系统运行层和 Linux 内核层^[1]. Android 有 4 大组件: Activity, Service, BroadcastReceiver 和 ContentProvider.

可扩展消息处理现场协议 XMPP (eXtensible Messaging and Presence Protocol) 是一种基于可扩展标记语言 (eXtensible Markup Language, XML) 的近端串流式即时通信协议. XMPP 中定义了 3 个角色, 客户端, 服务器, 网关. 通信能够在这三者的任意两个之间双向发生. 服务器同时承担了客户端信息记录, 连接管理和信息的路由功能. 网关承担着与异构即时通信系统的互联互通, 异构系统可以包括 SMS (短信), MSN, ICQ 等. 基本的网络形式是单客户端通过 TCP/IP 连接到单服务器, 然后在之上传输 XML. XMPP 协议的优点: 分布式, 安全, 可扩展, 弹性佳, 多样性, 分布式的网络架构^[2].

数据库系统 MySQL 所使用的 SQL 语言是用于访问数据库的最常用标准化语言. MySQL 软件采用了双授权政策, 分为社区版和商业版, 由于其体积小、速度快、总体拥有成本低, 尤其是开放源码这一特点, 一般中小型网站的开发都选择 MySQL 作为网站数据库^[3].

1.2 应用技术

在该 APP 中, 采用 JSON 格式的数据封装到 XMPP 协议里进行通信, 并用高德地图提供联想搜索和导航功能, 采用 GPS 技术获取车辆位置, 便于查找和远程召唤车辆, 并利用 Android 的 Service 功能保持 APP 后台运行, 然后采用极光推送平台进行消息推送.

JSON (Java Script Object Notation) 是一种轻量级的数据交换格式. 它是基于 ECMAScript (欧洲计算机协会制定的 js 规范) 的一个子集^[4], 采用完全独立于编程语言的文本格式来存储和表示数据. 简洁和清晰的层次结构使得 JSON 成为理想的数据交换语言. 易于人阅读和编写, 同时也易于机器解析和生成, 并有效地提升网络传输效率.

高德地图开放平台为 Android 系统所提供的服务包含地图、定位、导航、搜索、路径规划、室内地图^[5]等. 该 APP 通过加载高德地图 SDK 主要用到 POI 联想搜索和导航功能.

利用 GPS 定位卫星, 在全球范围内实时进行定位、导航的系统, 称为全球卫星定位系统, 简称 GPS^[6].

远程车辆定位功能调用手机的 GPS 和车机的 GPS, 来进行人与车的定位和导航.

极光推送, 英文简称 JPush, 是一个面向普通开发者开放的, 免费的第三方消息推送服务^[7]. 开发者只需在客户端集成极光推送 SDK, 即可轻松地添加 Push 功能到他的 APP 中. 该 APP 运用极光推送 API 实现车辆实时情况的消息推送.

2 分析与设计

2.1 需求分析

车辆远程控制 APP 的常规功能是注册登录功能, 其他功能是根据一些场景一一对应设计出来的. (1) 用手机控制车辆, 首先要知道控制哪辆车, 对应有绑定车辆功能; (2) 用户坐在自己家里想知道自己车的基本状况, 如车内温度, 空调开关情况, 车门是否上锁, 胎压胎温是否异常等等, 对应有查看车辆状态和故障报警功能; (3) 用户的车停在大型车库, 要去取车时反复寻找但找不到自己的车, 这时用远程找车功能, 可以快速找到爱车; (4) 有些停车位比较难停, 对于新手司机停车是个大问题, 对应用远程自动泊车功能; (5) 用户有时有事情又急于用车, 这时候希望车能够自行开到指定位置, 应运而生远程召唤功能; (6) 用户没有打开 APP, 但此时车辆有故障或者电量过低, 手机会收到推送消息进行提示; (7) 用户有时忘带车钥匙, 但需要打开车门, 此时就需要远程开车门功能; (8) 夏天天气炎热, 车停在路边导致车内温度过高, 为了方便用户能够随时上车, 可以使用远程开空调功能提前给车内降温. 还有提供联想搜索功能可以设置家庭和公司的地址, 提供导航功能等等.

与 APP 对接的后台监测系统需要完成以下几个任务: (1) 制定与 APP 的接口; (2) APP 用户注册和登录的账号, 用户名, 密码管理; (3) APP 版本的升级; (4) 管理车机 ID, 找到对应的车机系统; (5) 作为通信中转站, 转发 APP 与车机系统之间的通信数据, 并保存这些数据到数据库.

2.2 总体设计

用户通过 Android 客户端发送操控指令给后台, 后台接收指令并将指令转发给车机系统, 并在后台数据库保存指令消息. 车机系统做出相应的处理并将处理结果通过后台再传给客户端进行界面显示或进行通知栏推送消息等. 车机端安装一个车机控制系统, 该系

统会接收来自服务器的消息,然后解析出用户给的指令,通过CAN总线发送信号给车的XCU部件,然后XCU做出相应的控制操作,等待底层部件通过CAN总线回复成功或失败的信号后,该车机系统再把该信号封装成XMPP协议的消息回复给服务器,然后服务器解析出目的JID,然后再把消息转发给该DST JID.整体架构如图1所示.

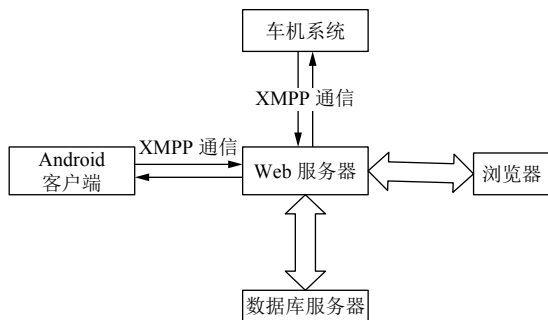


图1 整体架构图

2.3 Android 客户端 APP 设计

车辆远程控制 APP 采用 MVC 的架构设计模式,该模式可提高开发效率,降低模块间的耦合性. MVC 中的 M (模型) 是指获取到的 XMPP 协议所解析出来的 JSON 数据. 视图层是每个 XML 的布局文件 main_activity.xml 等,包括自定义的 view. 控制层是指 APP 开发中的各个 Activity 和 Fragment,如 MainActivity, ErrorActivity, ParkActivity, PositionSelectActivity, VehicleBindActivity, LoginActivity 等. APP 的主页面采用多个 Fragment 嵌套在一个 Activity 中的方式实现,使用 Viewpager 来控制多个 Fragment 的切换,增加页面切换灵活性.

车辆远程控制 APP 的主要功能有: APP 注册和登录,车辆绑定,车辆基本状态显示,远程开关车门,远程开关空调,远程寻车,车外自动泊车,车辆定位,车辆远程召唤和 OTA 升级等等.

APP 的主要功能结构图如图2所示.

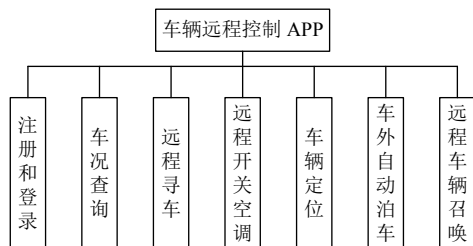


图2 车辆远程控制 APP 功能结构图

2.4 后台监测系统设计

后台监测系统功能结构图如图3所示.

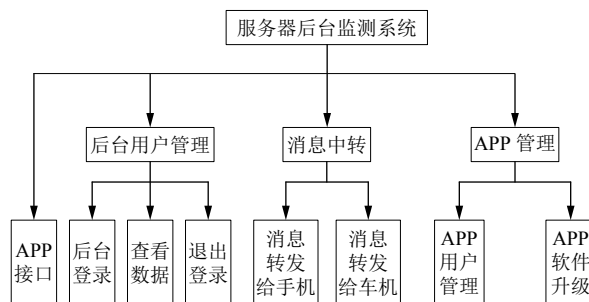


图3 后台监测系统的功能结构图

3 远程控制 APP 设计实现

3.1 注册和登录功能的实现

用户打开 APP 点击注册按钮,进入注册页面,进行用户名,手机号和密码的输入,然后通过 Okhttp^[8]网络框架进行服务器请求将数据上传到后台,后台系统判断该账号是否已经被注册过,如果已经注册过给出已注册提示,如果未注册过写入服务器,给出注册成功提示,然后页面跳转到登录界面,在登录界面输入账号密码进行登录,登录数据也通过 Okhttp 进行服务器请求,将账号和密码与后台数据库已经注册的账号密码匹配,如果存在并且密码正确提示登录成功,如果不存在提示未注册,如果密码不正确提示重新输入,登录后进入车辆绑定界面.在登录界面如果选中记住密码,则下次可以进入登录界面可以直接点登录,不用再重新输入.注册和登录流程图如图4所示.

3.2 主界面的实现

主页面由底部导航栏和车辆基本状态两部分组成,底部导航栏有3部分组成,分别是轮胎状态 Fragment,车辆基本状况 Fragment,远程控制 Fragment,3个 Fragment 可以用手指滑动进行切换.车辆基本状态页面主要显示车内温度,剩余电量,空调状态和车门状态,页面左上角是个人信息设置,右上角是故障报警.空调和车门锁是两个按钮事件,监听按钮点击事件,可以点击这两个按钮通过 XMPP 协议发送指令. XMPP 协议格式见表1.

轮胎状态页面显示胎温和胎压的数据指标,如有异常,对应的轮胎会亮红.远程控制页面有6个按钮事件组成,对应着6个功能,下面会一一详细介绍.主页面效果图如图5所示.

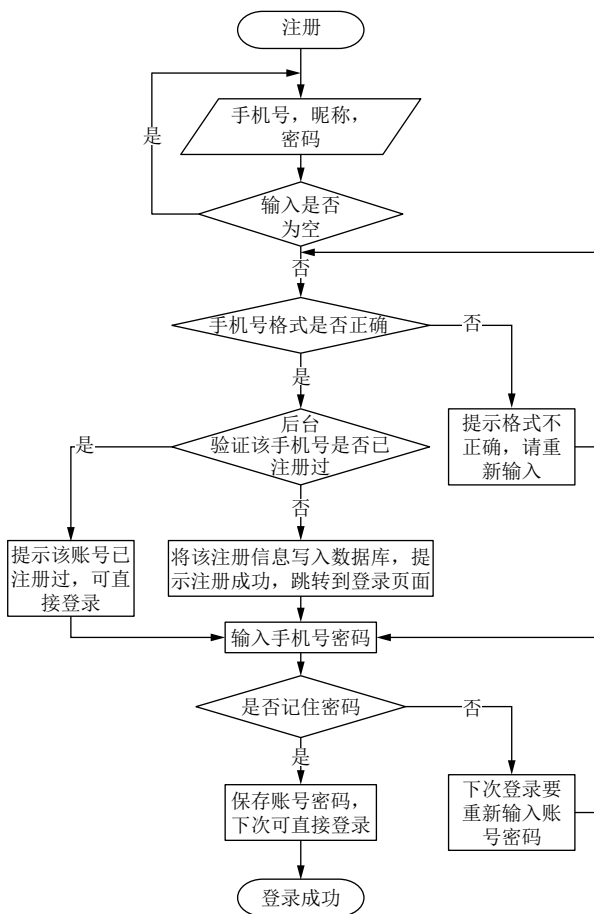


图4 注册和登录流程图

表1 XMPP 协议格式

类别	参数	说明
MsgHeader	DstAddr	username@serverIP
	SeqNum	命令消息序列号
	ServId	作为功能区关键词
	SrcAddr	carname@serverIP
	TimeStamp	时间戳
MsgContents	MsgType	消息类型
	MsgBody	消息内容

3.3 远程寻车, 远程开空调, 远程开车门

点击远程控制页面的远程寻车按钮, ButtonListener 监听到该点击事件, 即会通过 XMPP 协议发送寻车指令给后台, 后台解析该消息头, 找到车机 JID, 然后把 JSON 格式的 MsgContents 继续封装进 XMPP 消息, 发送给车机系统, 车机系统收到消息后就会控制车辆进行闪灯和鸣笛, 方便用户寻找. 远程开空调功能类似远程寻车, 点击远程开空调按钮, ButtonListener 监听到该点击事件, 即会通过 XMPP 协议发送开空调或关空调指令给后台, 后台解析该消息头, 找到车机 JID, 然后

把 JSON 格式的 MsgContents 继续封装进 XMPP 消息, 发送给车机系统, 车机系统收到消息后就会控制车辆进行开或关空调, 并且将操作结果返回给手机端在页面上呈现. 远程开车门命令与远程寻车和远程开空调功能类似.

3.4 车辆定位功能

打开主页面的车辆定位按钮, 会进入地图界面, 进入该界面同时, APP 会发送一条 GPS 的请求给车机系统, 等待获取到返回的车辆 GPS 信息后, 会显示车辆的位置在地图上, 并且会调用手机的 GPS 显示用户的位置, 此时点击导航到车, 会出现用户和车的路线规划, 用户可以选择导航到车的位置, 即会开始实时导航, 导航到车的位置后自动结束导航. 车辆定位界面如图 6 所示.

3.5 车外自动泊车功能

打开主页面的车外泊车按钮, 进入车外泊车页面, 同时会向车机请求车位信息, 车机系统会把车位的 4 个坐标点 $(x1, y1), (x2, y2), (x3, y3), (x4, y4)$ 发过来, 并且会有一个标记位记录该车位是否为空, 手机端收到车位信息后要把数据解析出来, 然后通过标记位选出空车位, 并记录下空车位的 4 个坐标信息, 然后再根据车位与车的相对位置, 在界面上自己绘制 view, 车在 view 的正中间, 然后以车为中心, 首先将车位坐标系进行转换, 转换成为手机坐标系, 然后再将车位的坐标点转化成像素, 然后与车的坐标进行比较, 车位的 x 坐标小于车的 x 坐标的位于车的左侧, 车位的 x 坐标大于车的 x 坐标的位于车的右侧, 车位的 y 坐标小于车的 y 坐标的位于车的上侧, 车位的 y 坐标大于车的 y 坐标的位于车的下侧 (比如, 车的像素坐标是 $(540, 960)$, 那么车位转换后的像素坐标为 $(480, 550)$, 这个车位就位于车的左上侧) 以此绘制车位图, 绘制完成后空车位可供用户随意选择, 用户触屏选择要停进去的车位, 然后 view 会监听该手指点击事件, 获取到选择的空车位编号, 将该编号组成 JSON 格式的数据封装到 XMPP 协议里面, 发送给车机, 车机系统收到车位选择消息后, 开始控制车辆开始自动泊车. 泊车成功后会推送成功的消息给手机端, 如果泊车失败, 车上的 360 摄像头会拍摄一张场景图发送给后台, 然后后台转发给手机端, 手机端下载图片后显示给用户. 车外泊车效果图如图 7 所示.



图5 主页面效果图



图6 车辆定位效果图

始控制车辆移动, 用户松开按钮即发送停止命令, 然后经过后台系统转发个车机系统控制车辆停止。



图7 车外泊车效果图

3.6 远程召唤功能实现

点击主页的远程召唤功能, 进入远程召唤页面, 如图8所示, 4个位置的按钮可以控制车辆前后左右移动, 点击按钮即发送移动指令, 后台监测系统收到前端APP发来的指令后, 进行解析, 消息Head部分解析出目的地JID信息, 然后将Body部分封装成一个新的消息, 将该消息转发给JID对应的车机系统, 车机系统开

3.7 远程状态查询功能实现

下拉主页进行刷新, 此时会触发APP后台会发送一条状态查询指令, 查询的状态包括车内温度, 空调开关, 车门开关, 车的剩余电量, 胎压胎压, 车辆是否有故障等. 服务器端收到这条指令后进行MsgType和MsgHeader的解析, 然后解析出车机JID, 将该指令重新封装成XMPP消息发送给对应的车机, 车机解析出指令, 然后通过CAN把指令发给对应的XCU, 通过对应的XCU收集到需要的状态, 然后把该状态封装成XMPP消息回复给对应的服务器, 再有服务器转发给对应的手机JID, 然后手机端会根据返回的状态在界面上进行

显示,如图5所示,如果车有故障,在图5的“我的汽车”界面右上角的角标会亮红色,并且显示故障的数量,然后点击该红色角标会显示具体的故障信息.



图8 车辆召唤功能实现页面

4 结束语

本文设计和实现的车辆远程控制 APP,方便用户随时随地的查看自己车辆状态信息.针对该 APP 的功

能进行场景需求分析,主要讲述 Android 系统手机端 APP 的设计以及各个功能的具体实现.最后该 APP 完成了真实的车辆控制原型系统,配合车机系统投入实际的应用当中,为用户提供了方便,提高新能源车的使用体验.

参考文献

- 1 秦晓凤. Android 系统架构与应用问题探讨. 信息通信, 2016, (8): 169-170. [doi: 10.3969/j.issn.1673-1131.2016.08.084]
- 2 石清超. 基于 XMPP 推送的 Android 个性化信息集成系统的研究与实现[硕士学位论文]. 青岛: 中国海洋大学, 2013.
- 3 王诚. 研究 JDBC 连接 MySQL 数据库技术. 网络安全和信息化, 2018, (12): 68-69.
- 4 仇小花, 秦栓栓, 邱果. 基于 WEB 开发中的 XML 与 JSON 数据传输格式研究. 信息技术与信息化, 2017, (4): 123-125. [doi: 10.3969/j.issn.1672-9528.2017.04.034]
- 5 甄文媛. 高德地图: 科技如何让出行更有温度. 汽车纵横, 2018, (3): 61. [doi: 10.3969/j.issn.2095-1892.2018.03.019]
- 6 胡风珍. 汽车 GPS 卫星导航全球定位系统技术研究与应用. 石油仪器, 2007, 21(4): 46-48, 51.
- 7 黄英伟. 整合极光推送和地图定位的智能手机打车应用研究[硕士学位论文]. 长沙: 湖南大学, 2016.
- 8 李群. 基于 OkHttp 的文件传输设计与实现. 电子技术与软件工程, 2018, (13): 180-181.