

# 移动泵站管理系统设计与开发

陈家健<sup>1</sup> 严良文<sup>1\*</sup> 陈善超<sup>2</sup>

<sup>1</sup>(上海大学机电工程与自动化学院 上海 200444)

<sup>2</sup>(北京亚控科技发展有限公司 北京 100086)

**摘要** 由于采用PC作为管理应用端,传统泵站管理系统无法做到随时随地监测设备信息,且无法将设备故障及时上报。针对这些问题,设计并开发“移动泵站管理系统”,除系统的移动端基于Android平台开发外,利用HTML5便于更新与维护的特点,通过混合开发,既解决纯HTML5无法调用底层硬件的问题,又解决原生软件开发更新困难的问题。运行结果表明:所开发的系统具有较好的易用性和便捷性。

**关键词** Android 混合开发 移动泵站管理系统

中图分类号 TP311.5

文献标志码 A

DOI:10.3969/j.issn.1000-386x.2024.07.003

## DESIGN AND DEVELOPMENT OF MOBILE PUMPING STATION MANAGEMENT SYSTEM

Chen Jiajian<sup>1</sup> Yan Liangwen<sup>1\*</sup> Chen Shanchao<sup>2</sup>

<sup>1</sup>(School of Mechanical and Electrical Engineering and Automation, Shanghai University, Shanghai 200444, China)

<sup>2</sup>(Beijing Yakong Technology Development Co., Ltd., Beijing 100086, China)

**Abstract** Due to the use of PC as the management application, the traditional pumping station management system can not monitor the equipment information anytime and anywhere, and can not report the equipment failure in time. In response to these problems, "Mobile Pumping Station Management System" is designed and developed. In addition to the mobile terminal of the system being developed based on the Android platform, the system utilized the feature of HTML5 that was easy to update and maintain. Through hybrid development, it solved the problem that pure HTML5 could not call the underlying hardware, and solved the problem that native software development and update were difficult. The operating results show that the developed system has better usability and convenience.

**Keywords** Android Hybrid development Mobile pump station management system

## 0 引言

近年来,基于移动互联网和手机的移动管理平台成为传统行业在产业互联网时代必不可少的一环<sup>[1]</sup>。传统的泵站管理系统基于桌面级应用开发,无法做到随时随地查看设备状态、水位情况;且多数管理企业对水泵维修报表的产生和上报还停留在人工处理阶段。

基于泵站管理企业的实际需求,开发移动泵站管理系统是十分必要的<sup>[2]</sup>。

## 1 系统概述

### 1.1 系统简介

某水泵管理公司为了更好地管理某一区域泵站,需要开发一个可及时监测各个泵站运行状态的移动泵站管理系统。

### 1.2 功能性需求

根据系统实际需求,系统划分为两类用户:普通用户与高级用户。

其中,普通用户需求如下:

(1) 泵站查询。用户可通过两种途径查看泵站状态:一类是在地图上显示泵站所在区域与状态,地图可缩放;另一类是通过选择不同区域查询,列表显示该区域内泵站位置与状态。

(2) 水位监测。用户可通过两种途径查看水位信息:一类是选择指定泵站,通过双折线图显示其 24 小时内内河与外河水位情况;另一类是通过选择不同泵站,以表格形式显示内河水实时水位。该查询需要提供历史水位回看功能。

(3) 雨量监测。用户通过列表查询指定区域实时雨量情况,该模块同样需要提供历史雨量回看功能。

(4) 天气预报。可向用户提供区域内 3 天天气预报。

(5) 预警。若水位或雨量超过警戒,则通过声音与振动提示用户及时查看并处理。

(6) 在线监控。通过泵站摄像头,在线监控水泵状况。

(7) 通讯录功能。根据不同单位查询其负责人联系方式。

(8) 巡检维修表格提交。用户可通过两种途径上传巡检表格至后台数据库:一类是通过扫描二维码跳转巡检表格,填写好巡检表格后提交并上传至后台,每个泵站拥有不同二维码;另一类是在软件内直接选择指定泵站填写表格,将表格上传至后台。现阶段巡检表格不定期更新。

对于高级用户,除了拥有上述普通用户的功能外,还有以下功能:

(1) 用户管理。可对普通用户信息进行查询,重置普通用户密码和删除普通用户账户。

(2) 对通讯录内容进行添加或修改。

### 1.3 非功能性需求

(1) 可靠性与稳定性。软件尽量不出现闪退、崩溃无响应等情况。

(2) 较高的交互性。根据用户习惯进行开发,保证软件的易用性,增加用户交互<sup>[3]</sup>。

## 2 系统开发环境

### 2.1 混合应用开发

HTML5 开发是基于 Web 技术进行的应用开发。HTML5 开发的好处是方便快捷,一次编写,可跨平台使用。但是由于 Web 自身的限制,H5 的移动端不能直接访问设备的硬件与离线存储,所以在体验上有局

限性。

Android 开发是利用官方提供的语言与工具进行开发,可直接对设备底层硬件进行操控,如摄像头的调用、离线存储、通知调用等。但是原生 Android 的可移植性差,无法在 IOS 端使用,且更新困难。

混合应用开发正是结合了原生应用开发和 HTML5 开发的技术特点,是取长补短的一种开发方式。底层基于原生代码构建,通过 WebView 或其他插件作为 HTML5 的容器,由“HTML5 云网站 + App 应用客户端”两部分构成。既可以增强软件与用户的交互性,又能在重要的功能上进行跨平台实时更新,易于维护<sup>[4-5]</sup>。

HTML5 开发,原生应用开发与混合开发的基本特点对比见表 1。

表 1 三类开发基本特点对比

对比项	HTML5	原生	混合
性能	慢速	快速	慢速
摄像头	无	有	有
通知	无	有	有
连接	主要为在线	在线与离线	在线与离线

对于该泵站管理软件而言,UI 布局无须经常性更新,只需要从数据库读取数据并显示,可使用原生 Android 开发将 UI 储存于本地,防止网络质量差时界面空白,增加用户交互性;此外,软件需要调用手机摄像头进行二维码扫描,通过调用手机通知,扬声器与振动马达对水位进行超量报警。所以主体软件客户端使用原生 Android 开发。

现阶段巡检表格不定时更新,比如改变表格内容、添加其他类型巡检表格等,且新增表格需要快速上线。巡检表格采用 HTML5 形式开发,使用 Table 将表格模板化,不但可快速新增表格,用户还可以不更新泵站软件,通过 WebView 在原生应用内直接操作新表格。

项目最终混合开发环境见表 2。

表 2 项目开发环境

类型	名称
开发环境	MySQL 5.5(数据库)
	Android Studio 4.1.2(软件端)
	WebStorm2020.3.1(HTML)
测试环境	Android 10(移动端)
	Android 7.1.2(移动端)
	Chrome 89.0.4389.90(浏览器)

## 2.2 混合开发语言

(1) Java 语言。Java 语言因其简单易用且可跨平台开发两个特征,被广泛地应用于软件开发中。

(2) JavaScript。JavaScript(JS) 基于原型编程、多范式的动态脚本语言,并且支持面向对象、命令式和声明式风格。

(3) HTML5。主要用于构建前端页面。相比于 HTML4,其为浏览器带来了更多的交互性<sup>[6]</sup>。

(4) SQL 语句。SQL 语句是一种数据库查询和程序设计语言,用于存取数据以及查询、更新和管理关系数据库系统。

## 3 总体设计

### 3.1 系统整体架构设计

移动端软件系统采用三层分层架构,从上至下分别为表示层、业务逻辑层、数据访问层。表示层内包含了软件的 UI 界面、检修表的内容、水位列表和折线图等可视化内容;业务逻辑层包含了软件各个功能与后台运行逻辑;数据访问层包含了对数据库的请求与访问和数据的接收。此外,云端数据库由数据访问层通过网络发送请求直接操控。系统拓扑图如图 1 所示。

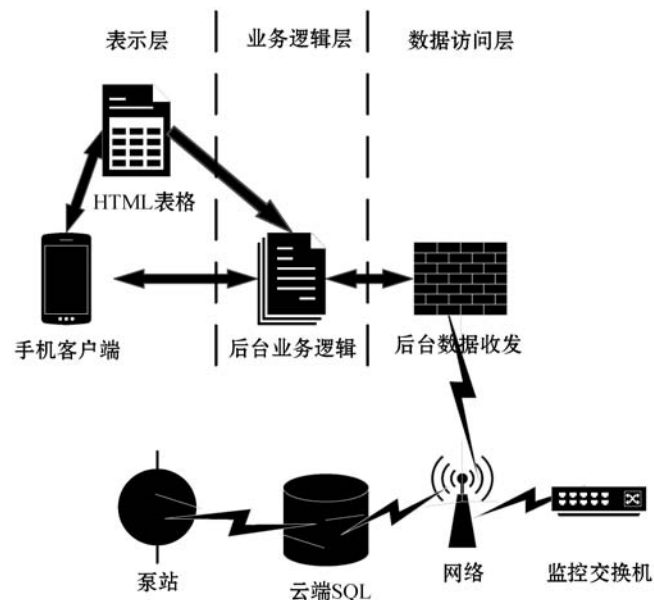


图 1 系统拓扑图

在该架构中,各层之间相互依赖。该架构移动端与后台交互过程以基本的水位查看作为例子:

(1) 用户通过手机客户端提交查看指定日期、指定泵站水位请求,软件将该请求提交到软件后台业务逻辑层。

(2) 业务逻辑层针对查看水位请求调用相应类,设置对应的数据库查询命令并发送到数据访问层。

(3) 数据访问层接收到查询命令后,便通过网络发送该命令操控云端数据库并从数据库获取相应查询结果。

(4) 数据访问层将结果返回给业务逻辑层审核,若查询错误,则客户端将 error 解析在表示层呈现给用户;若返回正确,则将数据,即水位信息结构化展现于用户。

### 3.2 模块设计

根据需求分析,得到系统各个模块之间的关系,如图 2 所示。

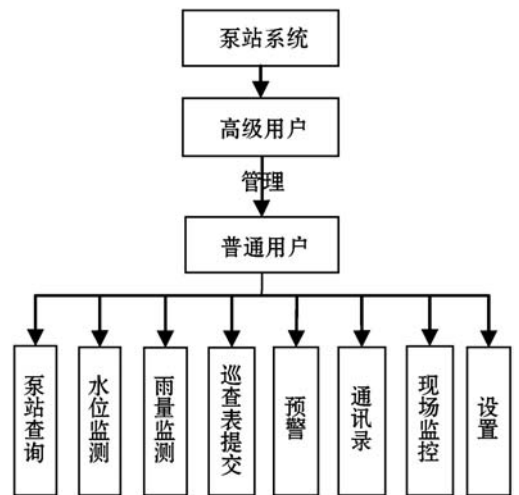


图 2 系统整体模块设计

### 3.3 数据库设计

为了让数据库更轻便,使数据的维护与更新更高效,该数据库将基于第三范式进行设计<sup>[7]</sup>。

根据图 1 所示的各个模块,在数据库中总共构建 7 张表,各个表内的详细内容见表 3。

表 3 数据库表信息

名称	说明
user	用户表。存放所有用户的基本信息,如用户名、密码、昵称、flag 权限。flag = 1 为高级用户,flag = 0 为普通用户
rain_level	站点雨量信息。储存历史雨量信息
water_level	站点水位信息。储存历史水位信息
site	泵站位置。储存泵站具体位置信息(区,镇,圩)
fix_ele	电气设备巡检信息表
fix_gate	闸门巡检信息表
abook	通讯录信息

user 表中高级用户将带上 flag = 1,普通用户 flag = 0,以此区分权限。将 site 表作为主表,以 site\_id(泵站编号)作为关关节点,与 rain\_level 表、water\_level 表、fix\_ele 表和 fix\_gate 建立连接。表的 ER 关系如图 3

所示<sup>[8]</sup>。

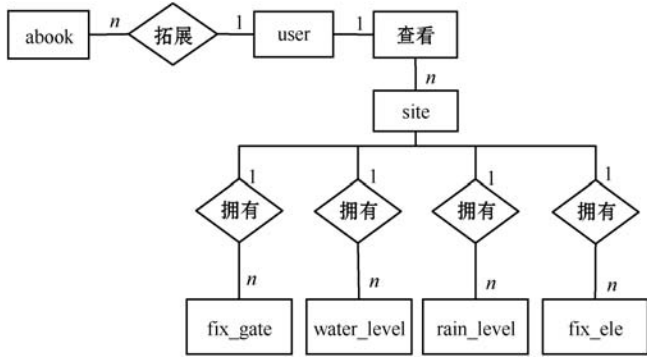


图3 ER 关系

### 4 系统实现

#### 4.1 用户登录实现

用户登录流程设计如图4所示。

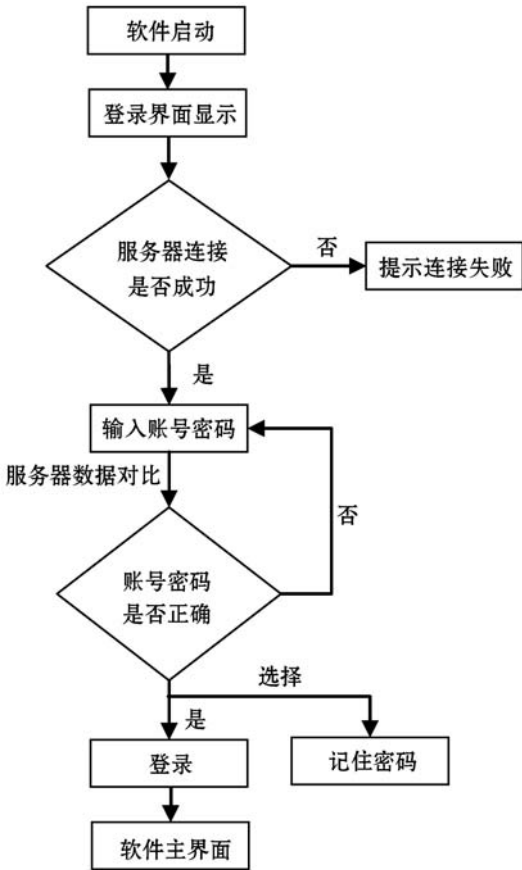


图4 登录模块设计

软件启动后自动连接数据库,若连接失败则使用 Toast 显示一则短消息在页面下方,提示用户软件连接服务器失败;当网络连接成功,用户输入账号密码后,软件使用二分法查询用户名,并使用 MD5 加密对比数据库内密码,若都正确,则进入软件主页面。

#### 4.2 主页面实现

天气预报功能使用和风天气 SDK,将天气预报功

能绑定于 main\_activity 内,使得用户可直观地看到管理区域当前天气情况。主页面内存在的按钮有泵站查询、水位监测、雨量监测、泵站巡检、现场监控、水位报警、通讯录和设置,除此之外在页面右上角存在扫码按钮以供扫描泵站二维码连接至对应数据表,上传巡检表格数据。主页面如图5所示。



图5 软件主页面设计

#### 4.3 泵站查询模块实现

该模块主要实现对管理区域内每个泵站的状态进行查询,该模块提供了两类查询方式:

(1) 地图模式,使用百度地图 JS API 进行地图生成,划分不同区域,将生成的 HTML 地图部署在 Tomcat 上,通过 WebView 进行访问与操控;使用 FrameLayout 增加纵向阵列显示当前所有泵站状态,如图6所示。



图6 地图显示泵站

(2) 列表模式。点击地图模式下方的列表按钮,从数据库读取泵站位置和状态,以列表的形式呈现,表内右下角提供刷新按钮。列表显示如图7所示。



图 7 列表显示泵站

#### 4.4 水位监测模块实现

水位监测主要实现对泵站实时水位与历史水位的查询,并以两种形式呈现:

(1) 折线图模式。读取指定站点指定日期的 24 小时内外河水位数据,使用 MPAndroidChart 开源 SDK 包,将数据通过折线图的形式呈现。当点击折线指定位置,折线上可显示 Mark 详细信息。由于折线图横向显示效果更佳,所以当处于折线图模式下,手机强制横屏;查询按钮下方为跳转水位列表按钮。折线图模式如图 8 所示。

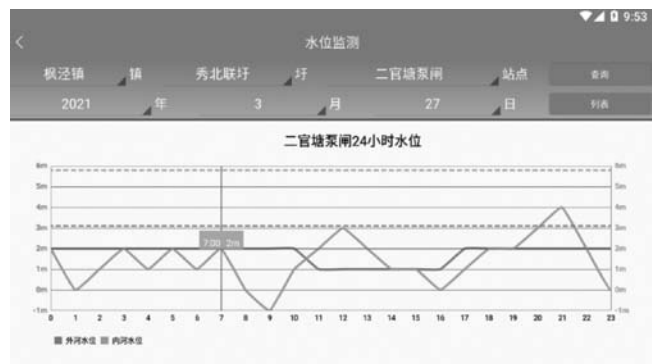


图 8 水位监测折线图

(2) 列表模式。将上述水位数据通过列表形式呈现;也提供列表模式下跳转水位折线图按钮。

#### 4.5 雨量监测模块实现

雨量监测与水位监测功能类似,同样提供两类呈现模式:折线图模式显示指定站点 24 小时雨量信息;列表模式以列表形式显示雨量信息。

#### 4.6 巡检模块实现

传统的巡检方式为员工例行到线下检查各个泵站状况,每个泵站填写对应巡检表,若发现问题,则在表内勾选不正常,再将表格带回公司统一处理。这样的巡检方式耗时耗力,且不能及时处理故障。

在该巡检模块中,用户可扫描线下泵站处的二维码,连接对应数据表,或者在软件内直接选择某区域指

定泵站连接数据表;再根据巡检表格的不同,逐份填写巡检表格。该模块还可拍照上传故障所在,使得维修人员能迅速定位泵站故障点。图 9 显示的是连接指定数据表后进行巡检表格填写并上传的流程<sup>[9]</sup>。

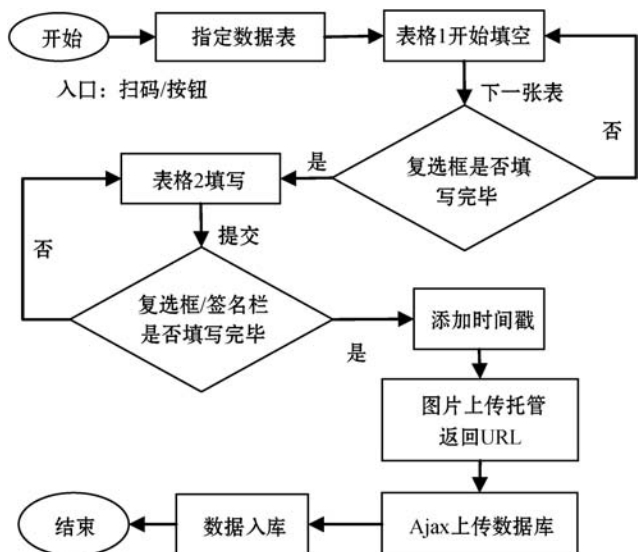


图 9 巡检模块流程

该移动平台大大简化巡检人员的工作量,提高了工作效率。巡检界面设计如图 10 所示,使用 WebView 展示 HTML 页面。

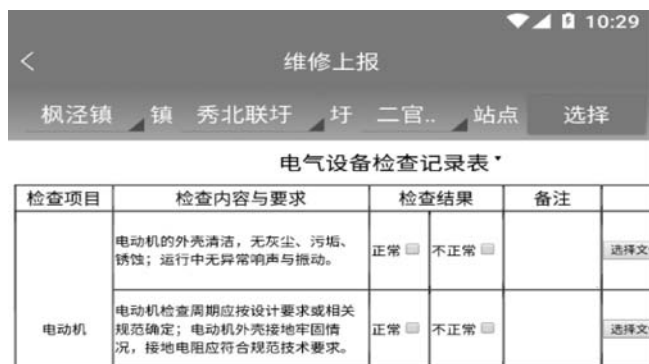


图 10 巡检表格界面

#### 4.7 水位预警模块实现

软件可选择后台常驻,若选择则软件在后台定时读取数据库水位信息,若有超低水位或者超高水位时,软件将调动扬声器与马达,通过声音和振动提示用户,对水位进行预警。

#### 4.8 通讯录模块实现

根据单位和部门,查询对应负责人员的联系方式。当账号为高级用户时,界面右上角存在添加按钮,且每个结果可点击进入修改其内容;当账号为普通用户时,右上角添加按钮将被隐藏,且查询结果不可点击修改。

#### 4.9 现场监控模块实现

由于现场水泵使用海康威视摄像头,软件通过海康 SDK 包对其进行配置,使用 SurfaceView 展示摄像头

内容。用户可以选择指定泵站,连接对应摄像头地址进行实时监控。

### 4.10 设置模块实现

在设置模块里,除了显示用户昵称、对用户手机号和密码进行修改外,还使用 Switch 控件对软件进行声音和震动的控制,并供用户选择软件是否常驻后台实时刷新数据以接收警报信息。设置界面如图 11 所示。



图 11 软件设置界面

## 5 关键技术

### 5.1 混合开发

该软件主要在巡检模块使用到了混合开发,其开发结构如图 12 所示。

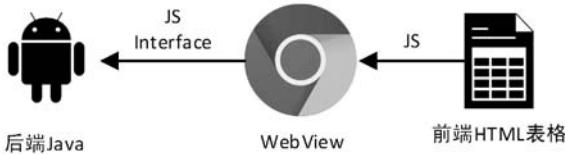


图 12 巡检模块混合开发

前端使用 HTML5 开发,基于 Table 规范化的巡检表格,可以以表格形式添加内容,以供未来表格内容的快捷增加,且通过 JS 与后端交互;中间件为安卓内置的 WebView 控件,用于显示并操控前端页面;后端部分为 Java 代码开发,用于设置 WebView、调用底层硬件、接收前端发送的数据。

数据交互部分:当用户填写好前端表格后,数据整合存放在数组内,通过 JS 传递到后端 JavaInterface 接口,后端接口接收数据后对数据格式化,发送到 SQL 语句中上传至云端数据库。

前端代码如下:

```
window.AndroidWebView.showInfoFromJs( sendData );
```

后端代码如下:

```
webView.addJavascriptInterface( new JsInterface( this ), "AndroidWebView" );
```

@ JavascriptInterface

```
public void showInfoFromJs( String getData[ ] ) { }
```

//前端与后端的方法名和传递数据类型需要一致

硬件交互部分:前端针对图片获取的部分,使用 JS 的 input 标签,属性 type 选择 file 即可通过 WebView 唤起后端的调用硬件代码;由于安卓版本碎片化严重,以后端需要针对不同的安卓版本做不同的硬件调用方法。

使用该混合开发的巡检模块在实际应用时体现了其混开的优越性:快速部署、无须更新、底层硬件调用良好。

### 5.2 Ajax 上传

在巡检模块中,将表格所填写的内容和放置于托管服务器内的图片地址一起发送到数据库内时,为了减少不必要的交互,节省流量,使用户的请求得到加速,并防止网页刷新导致数据丢失,该项目使用 Ajax (Asynchronous JavaScript and XML) 进行 GET 和 POST 请求。当用户点击提交按钮后,利用 Ajax 进行浏览器局部刷新,提示用户上传是否成功即可,无须对页面整体刷新<sup>[10]</sup>。

### 5.3 常驻后台

在安卓系统下,若不将软件加入白名单,当手机内存不足的时候会将软件进程杀掉以保证手机的续航与性能。该项目使用 AIDL (Android Interface Definition Language) 双进程守护。定义两个 Service,一个是后台操作的本地 Service,另一个是守护进程的远端 Service,其利用 AIDL 文件进行跨进程访问<sup>[11]</sup>。两个进程同时运行,如果有一个进程被系统杀死,另一个会将杀死的进程重新拉起,重写 onServiceDisconnected 方法,相互保护。

## 6 结 语

本文论述移动泵站管理系统的设计与实现,主要介绍系统需求、软件模块设计、数据库设计与重要模块的实现等内容。

系统实现了移动端和业务端的集成,将巡检表单与故障图片通过移动端随时随地上传数据库,很大程度上提高了员工的办事效率,实现离开 PC 办公;针对水位、雨量等情况,通过折线图与数字的形式进行显示,并将异常通过声音与震动的方式通知用户,让员工真正实现实时预知。未来将引入更多的模块与功能,比如使用 HTTP 转发推送请求的形式做到消息推

送<sup>[12]</sup>,将预警信息推送到安卓通知栏;还将考虑软件更深层次的前后端分离,从而提高软件的鲁棒性。

## 参 考 文 献

- [1] 张舒,刘道宇. 基于 Android 平台的移动办公软件的设计与实现[J]. 信息技术与信息化,2020(4):55-57.
- [2] 葛佩,李丛宇,张巧珍,等. 一种新型水务移动平台系统[J]. 净水技术,2018,37(11):17-21.
- [3] 肖文娟,王加胜. 基于 Vue 和 Spring Boot 的校园记录管理 Web App 的设计与实现[J]. 计算机应用与软件,2020,37(4):25-30.
- [4] 张志远. 基于 Android 混合模式开发的实习宝软件[D]. 北京:北京邮电大学,2016.
- [5] Oliveira W, Oliveira R, Castor F. A study on the energy consumption of Android app development approaches[C]//2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR),2017.
- [6] An J, Zhang L, Zhang X C, et al. The Effect of HTML5 on information security[R]. Science and Engineering Research Center;Science and Engineering Research Center,2017.
- [7] 陈长庆. 计算机基础应用数据库层次化结构设计[J]. 信息通信,2020(9):93-94.
- [8] Cheng X. Research on file upload based on HTML5[C]//International Conference on Service Systems & Service Management,2014.
- [9] 郭庆燕,张敏,杨贤栋. JQuery Ajax 异步处理 JSON 数据实现气象图片的显示[J]. 计算机应用与软件,2016,33(6):20-22.
- [10] 王汝言,蒋子泉,刘乔寿,等. Android 下 Binder 进程间通信机制的分析与研究[J]. 计算机技术与发展,2012,22(9):107-110.
- [11] 彭飞. 基于 XMPP 协议的 Android 消息推送的分析与设计[J]. 长沙大学学报,2018,32(5):41-43.
- [12] 吴昌政. 基于前后端分离技术的 web 开发框架设计[D]. 南京:南京邮电大学,2020.
- [EB]. MedRxiv:23286254,2023.
- [28] Rao A, Kim J, Kamineneni M, et al. Evaluating ChatGPT as an adjunct for radiologic decision-making[EB]. medRxiv:23285399,2023.
- [29] 彼特·李,凯里·戈德伯格,伊萨克·科恩. 超越想象的 GPT 医疗[M]. 芦义,译. 杭州:浙江科学技术出版社,2023.
- [30] Wang S, Zhao Z H, Ouyang X, et al. Chatcad: Interactive computer-aided diagnosis on medical image using large language models[EB]. arXiv:2302.07257,2023.
- [31] Huang L, Yu W J, Ma W T, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions[EB]. arXiv:2311.05232,2023.
- [32] OpenAI. Gpt-4 system card[EB/OL]. [2024-02-26]. <https://cdn.openai.com/papers/gpt-4-system-card.pdf>.
- [33] 中国政府网. 生成式人工智能服务管理暂行办法[EB/OL]. [2024-02-26]. [https://www.gov.cn/zhengce/zhengceku/202307/content\\_6891752.htm](https://www.gov.cn/zhengce/zhengceku/202307/content_6891752.htm).
- [34] Shu C, Chen B A, Liu F Y, et al. Visual Med-Alpaca: A parameter-efficient biomedical LLM with visual capabilities[EB/OL]. [2024-02-26]. <https://github.com/cambridgeltl/visual-med-alpaca>.
- [35] Thawkar O, Shaker A, Mullappilly S, et al. XrayGPT: Chest radiographs summarization using medical vision-language models[EB]. arXiv:2306.07971,2023.
- [36] Tang Y H, Liu F C, Ni Y S, et al. Rethinking optimization and architecture for tiny language models[EB]. arXiv:2402.02791,2024.
- [37] Wei J, Wang X Z, Schuurmans D, et al. Chain-of-Thought prompting elicits reasoning in large language models[C]//36th International Conference on Neural Information Processing Systems,2022:24824-24837.
- [38] Yao S Y, Yu D, Zhao J, et al. Tree of thoughts: Deliberate problem solving with large language models[C]//37th International Conference on Neural Information Processing Systems,2024:11809-11822.
- [39] Wang J Q, Shi E, Yu S, et al. Prompt engineering for healthcare: Methodologies and applications[EB]. arXiv:2304.14670,2023.
- [40] He J, Li P, Liu G, et al. PeFoMed: Parameter efficient fine-tuning on multimodal large language models for medical visual question answering[EB]. arXiv:2401.02797,2024.
- [41] Xiong H L, Wang S, Zhu Y T, et al. DoctorGLM: Fine-tuning your Chinese doctor is not a herculean task[EB]. arXiv:2304.01097,2023.
- [42] Gao Y F, Xiong Y, Gao X Y, et al. Retrieval-augmented generation for large language models: A survey[EB]. arXiv:2312.10997,2023.
- [43] Chen J W, Lin H Y, Han X P, et al. Benchmarking large language models in retrieval-augmented generation[EB]. arXiv:2309.01431,2023.

(上接第5页)

- [24] Li Y X, Li Z, Zhang K, et al. ChatDoctor: A medical chat model fine-tuned on a large language model meta-AI (LLaMA) using medical domain knowledge[J]. Cureus,2023,15(6):2045-2056.
- [25] Veen D, Uden C, Blankemeier L, et al. Clinical text summarization: Adapting large language models can outperform human experts[EB]. arXiv:2309.07430,2023.
- [26] 姜胜耀,李寅驰,薛万东,等. 基于大模型的多阶段出院小结质控方法研究[J]. 中国卫生信息管理杂志,2023,20(6):888-896.
- [27] Liu S, Wright A P, Patterson B L, et al. Assessing the value of ChatGPT for clinical decision support optimization