大 数 据 学 院

**课 程 设 计**

（2020-2021学年 第二学期）

**设计题目：**基于安卓的精美电话本的设计

**专 业**： 软件工程

**班 级:**  18软件本科1班

**组 别：**  第2组

**主 要 成 员：**  陈淳、郭森辉、魏凤萍、吴方圳

**指 导 老 师：**  大数据学院

2 0 2 1年6月25日

**目 录**

[1 引言 1](#_Toc1475)

[2 项目概述 1](#_Toc31279)

[2.1项目介绍 1](#_Toc11347)

[2.2 项目开发环境 1](#_Toc22744)

[2.3 小组人员及分工 2](#_Toc30441)

[3 需求分析 2](#_Toc11401)

[3.1 系统需求分析 2](#_Toc11295)

[3.2 可行性分析 2](#_Toc7211)

[3.3 系统流程图 3](#_Toc20240)

[4系统设计 4](#_Toc32503)

[4.1 系统结构设计 4](#_Toc24687)

[4.2 系统功能模块设计 5](#_Toc18350)

[5 系统功能及使用说明 6](#_Toc4463)

[5.1登陆功能 6](#_Toc3680)

[5.2联系人管理功能 8](#_Toc16094)

[5.3拨打电话功能 8](#_Toc21918)

[6 系统测试 8](#_Toc32287)

[6.1界面测试 8](#_Toc19981)

[6.2页面通用元素 9](#_Toc4835)

[6.3功能测试 9](#_Toc13230)

[7 总结 10](#_Toc7129)

[参考文献 11](#_Toc31326)

1 引言

随着带电子产品的快速发展，在收集用户大量激增的今天，人们需要记录管理的电话号码也越来越多。面对如此压力，简单的电话本将很难满足人们的需求，拥有一个良好的电话簿管理系统则是必须的。

2 项目概述

2.1项目介绍

当前时代，手机的普及以及互联网的高速发展使得万物互联，通讯也变成了人们每天必不可少的一件事情。相比于曾经的纸质电话簿，取代纸和笔，在手机上的电话簿显得更加方便快捷，随时随地记入亲朋好友的电话。在云存储地技术帮助下，不用顾虑丢失电话簿，永久备份电话簿。本小组的精美电话簿就是在这样的需求下，用C/S架构来开发的电话簿平台。

本平台面向所有用户，提供电话簿的使用。精美电话簿有用户登录、浏览联系人信息、向联系人发送短信、与联系人拨打电话、添加新联系人、设置白名单黑名单，查看短信等功能，该平台使用JAVAEE技术来开发后台Web API接口，并使用Tomcat作为服务器。该平台使用MySQL技术来连接数据库。

本平台现在能够满足了用户使用电话簿的要求，在安全性方面有一定的保障。随着技术的发展，为满足用户更深层次的需求，将进一步研发更新和扩展应用。

2.2 项目开发环境

电话簿拥有着更强大的功能，其开发主要包括用户后台数据库的建立与维护，以及用户电话簿的图形界面两个开发方面，对于前者服务器需要一个性能更优、完整性更强和具有一致性等特点的数据库，同时还不缺乏安全性。后者更需要应用程序更加具有美感，符合多数用户审美的设计界面，还需人性化的操作，简易使用的特点。

因此，本项目使用MySQL来建立后台数据，使用Tomcat来搭建Web API的运行环境，服务器使用Nginx来代理用户的请求，保证服务器的运行稳定。

硬件环境：

处理器：1核心

内存：2GB

软件环境：

操作系统：Window 10/Linux

服务器：Nginx

Java版本：1.7；

数据库：通讯云mysql数据库；

MySql图形界面管理工具：Navicat for mysql；

开发工具：MyEclipse Android Studio

2.3 小组人员及分工

陈淳为本项目的组长，主要负责分配任务、搭建系统框架、安排系统开发思路。郭森辉负责后端，数据的操作，逻辑控制，制作合理的接口供前端使用。魏凤萍负责前端开发，ui界面、运用mvvm框架、实现后端全部接口。吴方圳为产品经理，分析整个软件功能、测试软件、编写文档供开发人员进行操作。

如下表所示：

|  |  |  |
| --- | --- | --- |
| **姓名** | **职务** | **职责** |
| **陈淳** | 组长 | 分析程序，分解问题，形成总体设计开发思路，分析设计中用到的关键模块，收集资料与情报。合理分配任务，帮助组员实现功能解决bug。 |
| **郭森辉** | 后台开发 | 后端开发，制作后台各种接口，并提供接口文档给予前端开发。深入分析各个小问题，列出大纲，编写后端各部分程序模块，与产品经理对接。 |
| **魏凤萍** | 前端开发 | 前端开发，依据后端人员提供的文档实现后端提供的各种接口，制作优美的界面 |
| **吴方圳** | 产品经理 | 分析程序功能，并编写需求分析文档提供给前端与后端开发人员，测试软件并反馈问题 |

3 需求分析

3.1 系统需求分析

一个完整的通讯录管理系统应该包括:数据集模块，登录窗体模块，系统管理模块，主窗体模块，通讯录数据管理模块，信息管理模块，登录信息查询模块。本系统主要实现我们信息的管理，包括我们平时朋友信息的添加、修改、查询和删除，及一些辅助项目的管理，如登录系统用户管理和用户组管理。

3.2 可行性分析

**3.2.1工作负荷**

由于数据量越来越大，人工记录不一-定能适应目前的庞大数据量，工作负荷过大。

3.2.2费用支出

由于人工记录的工作负荷严重超载，而且易丢失，若投入的人力，设备，空间，材料等等与其他的一系列支持性服务越来越大，导致开发费用支出巨大，严重影响经济可行性，很不划算。

3.2.3人员设备

鉴于原有记录方式的技术性含量比较低，故不需要多少高技术人员的操作，只是由于数据量的日益扩大，所需要的操作人员数量不断增大。这一点已不再适应目前信息化时代的步伐。设备的科技含量也比较低，没有达到大量的计算机普及程度。

3.2.4局限性

经过严谨的分析，可知原有的管理方式存在很大的局限性，比如修改删除不方便，查询效率低等等- - 系列缺点，所有这些都明确了需要- -个新的适应新的信息化时代的系统。所以开发这样一个系统还是很有必要的。

3.3 系统流程图

本系统为B/S架构，整个项目前后端分离，前端HTML+CSS+JS制作用户页面，经过Tomat服务代理转发给客户端。其中大量运用AJAX技术，异步请求后端。经过Tomat服务器IP代理转发，自动寻找至Java EE程序。后端Java EE 提供前端所需要的接口，每个接口实现各自功能。数据库采用mysql，通过Java EE程序处理数据库中的表、关系，将数据以JSON格式转发至特定接口。当前端接受后端接口发送的Json数据后，自动编译，并转成页面相对应的数据。前端采用了MVVM思想，更合理的操作了数据。系统流畅，耦合度低，前后端完全分离。不仅可实现安卓端，通过更换前端UI还可实现PC端、Web端。后期我们将扩展项目，实现服务器集群，制作高并发服务器，转型Java spring cloud微服务时代。

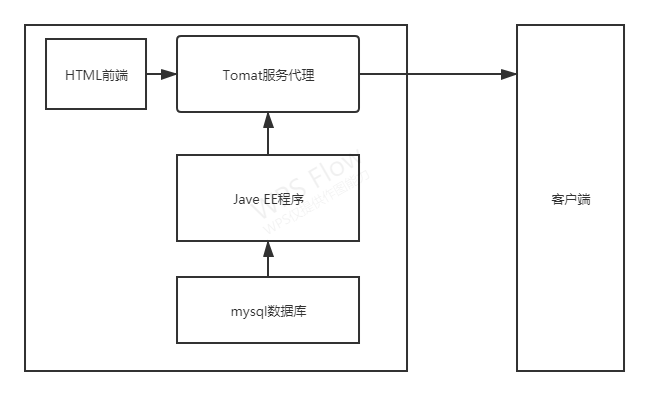


图3-1 系统整体框架

系统核心在于JavaEE后端处理，我们大致将后端分为4层：模型层、数据处理层、业务辑层、控制器层。其中视图层位于前端，控制器层与视图层使用WebAPI进行交互。



图3-2 系统核心后端框架

4系统设计

4.1 系统结构设计

精美电话簿是一类专业性的通话辅助功能系统，其总体结构可以作一下划分系统运行时，用户在应用子系统中工作，由应用子系统调用系统功能模块从而完成对系统数据的处理。

用户应用子系统是系统的用户界面。此层的缺失或划分不当，系统的用户友好性无从谈起。一般而言，应用子系统对应于用户某一需求的共同作业，此层面的设计与划分一定要从用户需求出发，面向电话簿的实际工作程序，以系统数据流图为基础进行。

从实用性入手，系统组织结构必须面向实际工作内容。为此，我们结合DBMS和GIS设计的概念和原理，将系统分为如下图所示的三个层次的七个子系统：各应用子系统都具有以下功能模块，其中包括属性数据库维护、用户数据库维护、数据检索查询、统计查询、矩阵判断、联系人关联分析模块。所有模块以GIS、DMBS类软件支撑并根据面向任务扩展产生。模块处理结果用文本、图片及声音三种方式输出，为用户电话簿使用提供决策依据。

4.2 系统功能模块设计

4.2.1登陆模块

用户填写账号和密码,提交到后台，后台获取到账号和密码后,将其发送到数据库中进行查询，查询结果如果为null,说明用户填写的账号或者密码有误,应该回到登录页面并提示用户重新输入，查询结果如果不为null,说明用户填写的账号和密码正确,将对应的账户信息共享到数据库中(在后面的请求,我们还需要继续使用当前登录的用户信息),然后跳转到应用的主页面。

4.2.2联系人管理模块

用户点击新建联系人后，输入联系人详细信息，应用储存联系人信息到本地文件夹，同时上传到数据库中备份数据文件。联系人建立后，用户可以点击修改联系人信息，修改联系人具体信息，后台数据库同步修改。顶部搜索框可以搜索联系人关键信息，包括但不限于联系人姓名、电话号码、手机号码和备注等。必要时用户可以删除联系人，应用将删除关于联系人的所有信息，数据库也将对该联系人打上删除标签。

4.2.3查看短信模块

应用调用android 的providers.telephony:SmsProvider:模块，调用content://sms/sent目录下的信息获取短信信息，包括但不限于收件箱、发件箱、草稿箱等。sms主要结构包括\_id：短信序号、thread\_id：对话的序号、address：发件人地址、person：发件人陌生人为null、date：日期，可以对日期显示格式进行设置、protocol：协议0SMS\_RPOTO短信，1MMS\_PROTO彩信、read：是否阅读0未读，1已读、status：短信状态-1接收，0complete,64pending,128failed、type：短信类型1是接收到的，2是已发出、body：短信具体内容。

4.2.4拨打电话模块

在按键中输入通话号码，目前电话本只支持一卡使用，不支持双卡双待手机，输入完毕号码拨打号码后，在Android Studio 中添加android.permission.CALL\_PHONE电话使用权限。申请拨打电话权限this.requestPermissions(permissions, REQUEST\_CODE\_CONTACT);输入电话号码后调用安卓原生电话拨打功能，拨打电话完毕后自动记入通话时间和通话详细信息，储存到通话记入当中。

4.2.5信息备份模块

可设定每天固定时间自动备份，系统自动将所有数据打包备份在服务器数据库的备份目录下，每天备份用户所有的备份文件，根据联系人文件大小可存储任意多个备份文件，平时用户只需经常从将手机端备份信息和联系人备份至数据库中即可增加保险系数。一旦发生意外，只须一份最近的备份文件就可还原恢复所有数据。

5 系统功能及使用说明

5.1登陆功能

用户凭在营业厅注册以后获得的账户和密码登录电话簿，登录后可以在个人信息修改界面中修改密码。如图5-1。

|  |
| --- |
| public static UserClass login(UserClass user) {  try {  //查询用户数据并返回  String username = user.getUsername();  String password = user.getPassword();  //进行查询  String sql = "select \* from user where username=? and password=?";  UserClass userClass = template.queryForObject(sql,  new BeanPropertyRowMapper<UserClass>(UserClass.class),  username, password);  return userClass;  } catch (Exception e) {  return null;  }  }  public static int register(UserClass user) {  /\*\*  \* INSERT INTO USER VALUES(  \* DEFAULT,'李斯',16,FALSE,'lisi','123','156986659865',DEFAULT  \* )  \*/  try {  String sql = "insert into user values(DEFAULT,?,?,?,?,?,?,DEFAULT)";  int update = template.update(sql,  user.getName(),  user.getAge(),  user.isGender(),  user.getUsername(),  user.getPassword(),  user.getTel());  return update;  } catch (Exception e) {  return 0;  }  } |

|  |  |
| --- | --- |
| -6779fb2283086039_edit_552831956367206 | 4e6b7f8a4029c9_edit_552823784660957 |
| 图5-1 用户登陆界面 | 图5-2 联系人管理界面 |

5.2联系人管理功能

在联系人管理中，可以编辑已有联系人的信息或者新建联系人。如图5-2。

5.3拨打电话功能

可在拨号界面输入号码后进行联系，或者在最近通话记录中快捷拨通想要联系的对象。如图5-3。

|  |  |
| --- | --- |
| 73e2eab151f1d2f7_edit_552814844344292 | -85f29f25bb87f7e_edit_552806691344293 |
| 图5-3 拨号界面界面 | 图5-4 短信查看界面 |

5.4查看短信界面

在通话记录界面中可以快速查看最近的通话短信。如图5-4。

6 系统测试

6.1界面测试

|  |  |  |
| --- | --- | --- |
| 测试内容 | 测试点 | 是否通过 |
| 页面显示 | 1、手机窗口标准或最大时页面元素显示是否正确，是否美观，窗口大小变化时页面刷新是否正确;  2、手机显示屏是宽屏或标屏下页面元素显示是否正确，是否美观;  3、用户常用的几种分辨率”下页面元素显示是否正确，是否美观。  4、字体的大小要与界面的大小比例协调，通常使用的字体中宋体9-12较为美观，很少使用超过12号的字体。  5、前景与背景色搭配合理协调，反差不宜太大，最好少用深色，如大红、大绿等。  6、页面弹出式提示界面必须大小合理，布局美观，符合系统风格。 | 通过 |
| 页面布局 | 1、布局要合理,不宜过于密集，也不能过于空旷，合理的利用空间。  2、相关页面元素的外形是否美观大方，大小是否合适，位置和页面的风格是否协调。  3、页面相关说明性文字的位置是否正确合适，按键定位在需说明的控件.上时相关，提示信息位置是否合理。 | 通过 |
| 页面风格 | 1、同一系统中不同页面的整体风格是否一致，是否美观; 2、各页面背景、色调是否正确，是否美观，是否适合应用环境。  3、主色调要柔和，具有亲和力与.磁力，坚决杜绝刺目的颜色。 | 通过 |

6.2页面通用元素

|  |  |  |
| --- | --- | --- |
| 测试内容 | 测试点 | 是否通过 |
| 输入框 | 1、不允许输入字符或汉字，不允许输入特殊字符;  2、应该根据允许输入的数据范围锁定输入长度;  3、综合运用等价类和边界值分析的方法确定具体测试  数据。 | 通过 |
| 按钮 | 1、按钮应该美观，大小合理，按钮上的内容位置应居中; 2、按钮和整个页面风格保持一致，布局位置合理;  3、功能操作的确认、重置、取消按钮应该在所有输入控件下方的合适位置;  4、鼠标指针移动到按钮上时应该自动变为手形(也可同时变化按钮背景色)。 | 通过 |
| 输入域 | 1、输入域要美观、漂亮，大小合理，和整个页面的布局融为一体;  2、输入域中输入内容时需提供自动换行功能，超出输入域长度或宽度时需要有滚动条可以操作。  3、输入域中文字的默认字体、大小、颜色需和整个页面保持一 致。  4、输入域中可输入内容应有长度限制，综合运用等价类和边界值分析的方法确定输入内容长度进行测试。 | 通过 |
| 导航 | 1、导航按钮风格和应用系统的页面结构、菜单、链接的风格是否- -致;  2、图片按钮导航或按钮导航应该可以准确切换到对应功能;  3、鼠标置于导航按钮上时应该显示成特殊的鼠标指针，且导航按钮应该高亮显示。 | 通过 |

6.3功能测试

|  |  |  |
| --- | --- | --- |
| 测试内容 | 测试点 | 是否通过 |
| 新增 | 1、新增功能应该不允许新增对应数据表主键内容重复的数据;  2、新增功能是否正确保存数据到对应的数据表中的正确字段;  3、新增功能不会影响数据库中已经存在的数据。  4、新增成功或失败都应该反馈准确的提示信息。  5、新增时应该自动处理掉输入内容两端的空格。 | 通过 |
| 修改 | 1、修改功能是否正确修改数据库中对应表的对应字段的数据;  2、修改功能应该不允许修改数据库中对应表的对应记录的主键数据;  3、修改功能不会影响数据库中与对应修改数据无关的数据，不会新增数据(除非新增处理是作废原记录并新增记录)。  4、修改成功或失败都应该反馈准确的提示信息 | 通过 |
| 删除 | 1、删除功能操作时必须提供删除确认步骤;  2、删除功能会正确删除数据库中对应的记录;  3、删除功能不会删除删除数据以外的任何数据。  4、删除成功或失败都应该反馈准确的提示信息。  5、根据条件删除数据的功能必须是精确条件删除。 | 通过 |
| 查询 | 1、查询功能需要区分实现精确查询和模糊查询功能;  2、查询功能需自动处.理输入内容两端的空格;  3、模糊查询需屏蔽掉SQL语句中用到的通配符;  4、查询效率应该可以符合平常使用要求。 | 通过 |

7 总结

本次Android及JavaEE Web Api应用开发的实践加强了我对Android和Java Web Api的深入理解，其中包括对界面的基本操作，Java代码的熟练，数据库的语句查询、服务器的接口传输。对所学的知识有着更加深入的理解。虽然在项目的制作过程中有着许多困难，但是通过团队合作配合，一往无前，无所畏惧，使得我们成功地克服了困难。包括在Android Studio、Tomacat Java jdk的安装与环境配置，在后期软件的制作过程中遇到了许多困难，但通过互联网和同学的帮助得以解决。我们只有不断地学习才能有提高，但是只有实践才能真正的将所学知识用在程序之上，才能有着正真的提高。

**参考文献**

[1]尚宇, 储妮晟, 雷斌. 基于BREW平台的手机电话簿的设计与开发[J]. 西安工业大学学报, 2006(04):345-348.

[2]侯万春. 具有电话簿自动更新功能的移动电话终端及其实现方法: CN, CN101399872 A[P].

[3]姜默涵, 陈榕, 黄玉坤. Elastos智能手机动态多UI电话簿排序的实现[J]. 计算机技术与发展, 2008, 018(001):196-198,201.

[4]张宁. 蓝牙车载免提应用中电话簿同步系统设计与安全机制研究[D]. 上海交通大学.

[5]访问动态电话簿的方法和设备: CN, CN1581896 A[P].

[6]郭斌. 安卓手机应用软件的取证研究[D]. 重庆邮电大学.

[7]李阳波. 电话号码簿[J]. 杂文选刊:上半月, 2006.

[8]张晋平, 李铁兵. 可视图文电话号码簿系统[J]. 数据通信, 1997, 000(001):28-33.

[9]孙开元. 情系电话簿[J]. 特别健康, 2017, 000(012):56-57.

[10]姜善敬. 电子电话簿的书签标记方法和装置:, CN 1383313 A[P].

[11][美] Jeffrey R. Shapiro 周之,黄玫译. SQL Server 2000参考大全[M].北京：清华大学出版社,2002.

[12]史九林.数据库概论[M].西安:西安电子科技大学出版社,1988.

[13]王向辉, 张国印, 陈洁.Android应用程序开发[M].清华大学出版社, 2010.

[14]张利国, 代闻, 龚海平编著.Android移动开发案例详解[M].人民邮电出版社, 2010.

[15]李燕.JAVA编程语言在计算机软件开发中的应用[J].计算机产品与流通,2020(06):13+27.