目录

[1.数据库设计 2](#_Toc24795204)

[1.1. 登录和注册的本地测试数据库设计 2](#_Toc24795205)

[1.2. 人脸数据库的建库 3](#_Toc24795206)

[2. 单元测试设计 3](#_Toc24795209)

[3. 接口设计 4](#_Toc24795210)

[3.1 百度人脸识别的接口设计 4](#_Toc24795211)

[3.2.本地sqlite接口的设计 5](#_Toc24795214)

[3.3.本地数据上传的通信接口 5](#_Toc24795219)

[4. 类设计 5](#_Toc24795222)

[5. 软件安装及使用说明 6](#_Toc24795224)

1.数据库设计

* 1. 登录和注册的本地测试数据库设计

进入程序页面以后首先要求用户登录和注册，对于登陆和注册输入规定了一些输入约束。如果没有登录，或者登录名错误，或者登录的密码错误，那么都是不可以登录进去的。需要用户更正自己的用户名或者密码。如果用户没有注册，那么还需要用户去注册页面注册，用户注册后填写自己的相关信息。

我们建立了一个user表来存储用户的注册信息。

|  |  |  |  |
| --- | --- | --- | --- |
| User | Name | Email | Password |
| 1 | libai | @110.com | 520 |
| 3 | \*\*\*\* | GHKJL@FGH.com | FGHJK |
|  |  |  |  |

User表存储用户的注册信息，各个属性为：name，email，password。

在后来的设计中，为了简化数据库的存储和查询，将其中的email属性删除。只保留name和password两个属性。

在新的版本测试中，为了使得查询更加简单，将name和email捆绑在一起，用标识符“，”隔开。最终的数据存储，按照标识符+name+标识符+password+标识符来表示。标识符采用“，”。

例子：

[，lijisfan，1808sdf909@#89.com，sfsffkd，](mailto:，lijian，1808032909@#89.com，lksjlfkd，)

[，lijian，1808032909@#89.com，lksjlfkd，](mailto:，lijian，1808032909@#89.com，lksjlfkd，)

以上就是登陆和注册的本地数据库的建库操作。

* 1. 人脸数据库的建库

人脸搜索功能中，数据库中手动上传了20位明星的人脸图像，在百度智能云上建立了人脸数据库去搜索匹配，最终得出相似度。没有建立个人的人脸数据库。

人脸对比功能中，从人脸数据库中选择两张图片或者用拍照功能获取两张图片进行对比，进行相似度对比。

1. 单元测试设计
   1. 登陆和注册的业务逻辑测试

我们测试了登陆和注册的逻辑功能的正确性。用户登录提示“是否已经注册”，用户注册，用户填写密码和用户名，核查用户的账号正确性，登录操作。

测试了各个部分的逻辑功能。首先是注册页面的逻辑测试。

用户点击注册的按钮后更换到注册的页面，注册的页面要求用户输入自己的name，password，确认password，email来完成注册。注册后点击注册按钮后，完成数据上传，并且返回主界面。

用户在主界面输入自己的账号信息，登录到main窗口。

* 1. 登录和注册的本地数据库测试

测试了本地数据库的登录和注册的数据库的功能。

测试了登录的时候验证用户用户名是否为空，如果用户名为空则要求用户去注册。

测试了登录的时候在数据库中查询用户的账号信息的功能。查询用户的name+password，如果用户的数据在数据库中被查询到，则允许用户登录到main窗口。

测试了注册页面重复注册的功能，我们并未提供重复注册检查的功能，因此当用户重复注册的时候，用户的信息也会重复的被插入到数据库中。但是只要数据库中存在一条用户的信息，name在登录的时候就会在数据库中被搜索到。

我们考虑到了这个应用的应用场景随着规模的扩张，对数据存储和搜索的要求。因此当规模扩大时，将会加入数据库的消除冗余的规则，来降低数据库的冗余度。

* 1. 人脸识搜索、识别和检测的测试

测试了人脸的上传、识别、搜索和检测的功能

测试了人脸图片的上传功能，人脸数据的上传，利用bitmap和hbase64来实现图片数据的传输和通信。

测试了百度人脸识别的接口，将其识别的结果返回到前端测试。

测试了百度人脸识别接口的可用性，我们在百度的人脸库中手动上传了20张人脸图片，并且在前端上传图片进行人脸的识别。识别的结果返回 了人脸识别的分类和人脸识别的相似度。人脸对比功能中，识别两个图片，将两张图片得相似度返回前端。人脸监测中将将识别到图片性别、年龄、心情返回到前端。

3. 接口设计

3.1 百度人脸识别的接口设计

比对两张图片中人脸的相似度，并返回相似度分值。用于人脸搜索、识别功能，比对多张图片中的人脸相似度并返回两两对比的得分，可得判断两张脸是否为同一个人的可能性。将得分最高的图片信息返回到前端进行输出。

检测图片中的人脸并标记处位置信息，展示出人脸的核心关键点信息，及150个关键点信息，可以展示出人脸属性信息，如性别、年龄、心情。

3.2.本地sqlite接口的设计

使用了本地数据库的sqlite的接口来实现数据库数据的存储，插入和查询；

使用了query接口来实现数据库的查询；

使用了put函数来实现数据的插入；

使用来了index来实现数据库的匹配，从查询的数据信息中匹配用户的账号信息是否存在的信息。

3.3.本地数据上传的通信接口

使用了bitmap和base64来实现数据的转化

使用了okhttp来实现数据的通信和传输

4. 类设计

创建了MainActivity，用来记录拍照后的照片文件的地址，查看照片的view，以及实现各个button之间的照片上传下载等页面跳转。

onRequestPermissionsResult方法去请求用户允许访问读取相册操作，利用actionName和Extra,启动《相机Activity》，相机拍摄完成后利用ContentResolver,查询临时文件，并使用BitMapFactory,从输入流中创建BitMap，进行sdcard的读写请求。打开相册，进行相关选择。将所选择的照片使用了bitmap和base64来将图片转码上传。上传到人脸数据库后，调用百度API接口实现人脸识别、搜索和检测功能。

5. 软件安装及使用说明

执行该软件的pc端必须是安卓机，下载安装包后根据系统提示安装即可。进入主页面后首先进行用户的登录和注册，在登录和注册的部分根据约束性条件可以实现用户的校验，通过匹配数据库中账户信息来判断其账号信息的正确性，从而来对用户的登录进行控制。

我们的软件可以实现用户的人脸识别、搜索和检测功能，用户拍照或者从相册选择，上传图片，就可以通过百度人脸识别的接口来识别用户的人脸后，与人脸数据库中的图像进行。同时也可以实现检查用户人脸的准确度，来和人脸数据库进行相似度匹配，来确定用户的人脸特征。

我们的软件在功能上实现了人脸的建库，可以对人脸数据库进行建立，插入人脸的数据和图像。通过在百度人脸的项目平台上插入人脸图像和人的数据信息，就可以建立人脸的数据。

我们的软件在功能上实现了人脸的搜索和检验，可以通过上传人脸的图片来对人脸的匹配度进行搜索，找到和目标人脸最为相似的人脸数据。