## 数据库设计：

Mongodb:

Schema：userSchema

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key | name | group\_id | user\_id | Img\_path | count |
| typer | String | String | String | String | Number |

## 单元测试

没有单元测试：

Android里的用Test测试，对每一个请求做了测试（最后都删除了）

Express中的测试直接npm start运行测试的，大部分测试代码也没有留下了

Mongodb的测试连接上后也是直接运行的

## 类设计

Android中

Activity：

PostImageActivity：上传底库图片（拍照或者相册选择照片）

ShowResultActibity：上传图片显示匹配的结果（拍照或这相册选择照片）

Express中

Router：

Index.js：分为两个：/addFace和/searchFace

/addFace：android发出添加人脸底库请求到Express，router用axios调用百度ai发送添加请求，并把图片存储到本地文件夹中，向mongodb中插入userSchema模式的数据

/searchFace：android发出人脸匹配请求到Express，router用axios调用百度ai发送添加请求，并把图片存储到本地文件夹中，根据返回的user\_id修改mongodb中相应的userSchema模式的数据

Node.js中

users.js

将对数据库中users的Schema的操作封装到users.js，分为addUser和updateOne两个function

## 本地接口设计

无

## 网络接口设计

Express的两个POST请求：

/addFace：添加人脸库

/searchFace：底库人脸寻找

百度AI的三个请求：

/oauth/2.0/token：获取Token

/rest/2.0/face/v3/faceset/user/add：人脸库添加

/rest/2.0/face/v3/search：人脸搜索

/rest/2.0/face/v3/multi-search：人脸搜索（M:N）---项目写了，但未实际应用

## 软件安装

1.阿里云上安装mogodb，并fork进程

2.将xj\_express01文件夹放入阿里云中，运行npm install安装依赖包，npm start运行Express项目。

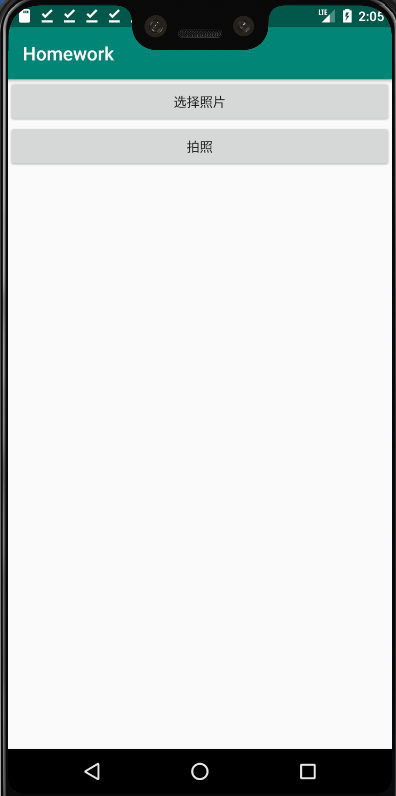
3.将apk安装到手机即可运行

## 使用说明

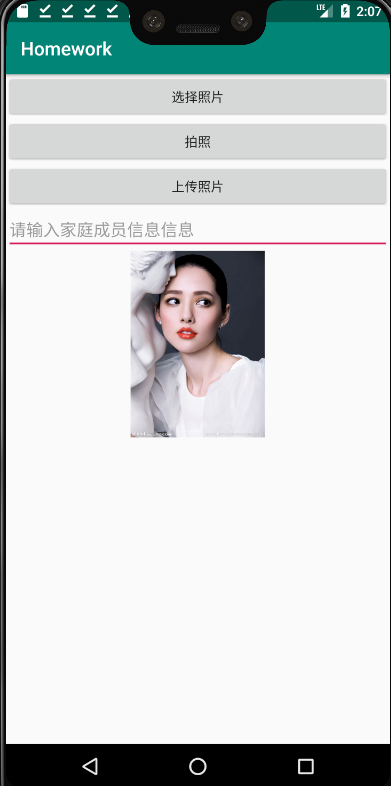
主界面：



第一个功能：1.建立家庭关系，进行添加底库照片：



2.点击选择照片是从相册中选取，点击拍照是通过摄像机选取照片，在选取照片后家庭成员信息和上传按钮会弹出



第二个功能：识别家庭成员：上传照片，寻找底库，操作同建立家庭关系