1 项目介绍

1.1项目背景

2.目的及意义

3.需求分析

3.1可行性分析

登入验证（学生，老师，管理员）；

学生：提交图片（修改图片）、修改密码、查看到课情况

老师：选择课程、拍照功能、提交照片、返回结果（统计结果：到课率等）、手动修改签到记录，添加删除学生信息

管理员：对数据库管理

4.技术方案（开发环境）

安卓、java、Windows、C++（seetaface）、Tomcat、Ubuntu

5.系统概要设计

5.1概述：  
本系统将实现在Android平台下的人脸识别考勤功能。系统将实现采集照片，人脸识别，考勤统计等功能。

本系统需要实现的功能：  
1.考勤人员和被考勤人员的注册与登录。

2.被考勤人员可上传自己的照片和相关身份信息，服务器将接收信息并存储在数据库中。

3.考勤人员在Android客户端上传照片，系统能识别照片中的人脸并计算出每张人脸的特征值。

4.系统从服务器获取被考勤人员的照片和相关信息，在系统中计算每个考勤人员照片上的人脸的特征值，与3中计算的特征值进行一一比较，得到考勤的结果。

5.根据前面的结果统计考勤信息并传回服务器进行存储。

6.考勤人员和被考勤人员均可登陆系统查看相关的考勤信息。

另外为了维护数据库和服务器的安全与稳定，需要管理员在Web端对整个系统进行相关管理。

5.2系统整体设计：

考勤系统需要在Android客户端完成考勤任务，同时与服务器进行交互，完成对数据库的相关操作。

同时管理员需要在Web端对服务器和数据库进行维护管理。



考勤人员

无线网连接



被考勤人

Android客户端

服务器





Web端

数据库

管理员

考勤系统的整体架构如图所示，考勤工作在Android客户端完成，管理员对系统的维护和管理在Web端完成。

Android客户端开发架构：

Android端采用C/S架构，客户端一方面负责与用户的交互，提供界面，另一方面负责与服务器交互，使用Socket进行数据交换。服务器负责与数据库之间的数据交换，对数据库进行增删改查等操作。体系结构图如图所示：

客户端

操作

考勤用户

登录注册

显示

数据库

服务器

考勤模块

操作

信息上传模块

被考勤用户

显示

用户使用客户端进行相关操作，客户端会进行相应的应答，并根据请求向服务器发送相关信息，通过服务器操作后台数据库。

Web端开发架构：

Web端采用B/S架构，管理员通过浏览器向服务器发送请求，服务器响应后管理员可对数据库进行管理和维护。

请求

数据库

服务器

浏览器

管理员

应答

5.3 Android客户端工作流程图

开始

N

注册

是否注册

Y

登录

被考勤人员

考勤人员

选择考勤目标

N

上传照片

是否已上传照片

Y

上传照片

查看自己的考勤信息

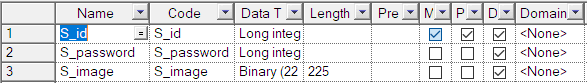
查看全体考勤信息

在此系统中，人脸的特征提取和考勤信息的统计均在本地完成，获得考勤信息后，上传到服务器保存到数据库，以便进行查看。用户进入系统后系统会根据不同的身份信息显示不同的界面，用户可进行相关的操作。考勤用户可以选择数据库中已保存的考勤目标，然后上传照片，系统会计算上传照片中人脸的特征值然后从服务器获取考勤目标的图片并计算考勤目标的人脸特征值，然后进行比较得到结果。被考勤人员上传自己的照片，服务器获取照片信息保存到数据库中。

5.4 数据库设计

数据库主要分为以下几个部分：学生信息，老师信息，课程信息，课程考勤的日志信息。

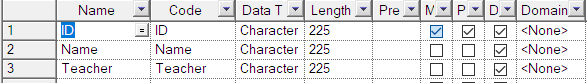
学生信息包含学生ID，姓名和照片; 主键是学生ID：



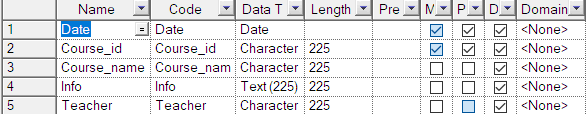
照片在数据库中用二进制的方式存储。

老师信息包含ID和姓名主键是老师ID：  


课程信息包含课程ID，课程名称，授课老师；主键是课程ID



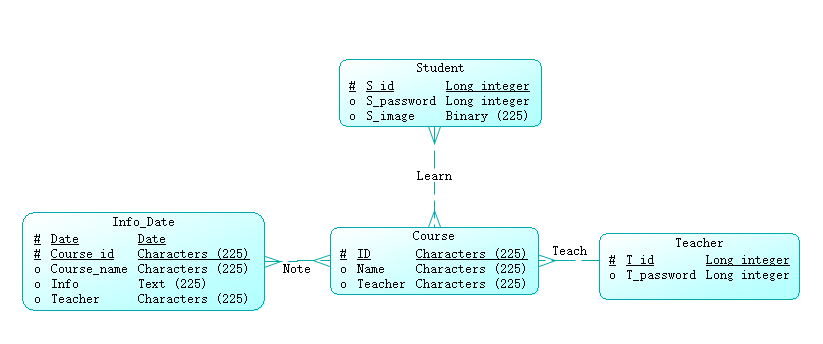
日志信息包括日期，课程ID，课程名称，授课老师，考勤信息



考勤信息用文本格式保存

在这个数据库中，学生与课程之间是many-many的关系，老师与课程之间是one-many的关系，日志信息和课程之间是many-many的关系.

E-R图如下所示：



7.时间安排表

2018.11-2018.12：技术调研

12-2019.1：数据收集、前端设计和实现

1. - 2. 各功能模块的实现
2. - 3. 优化和文档撰写

8.分工安排

安卓前端开发：

服务开发和搭建：

人脸识别技术的实现：