

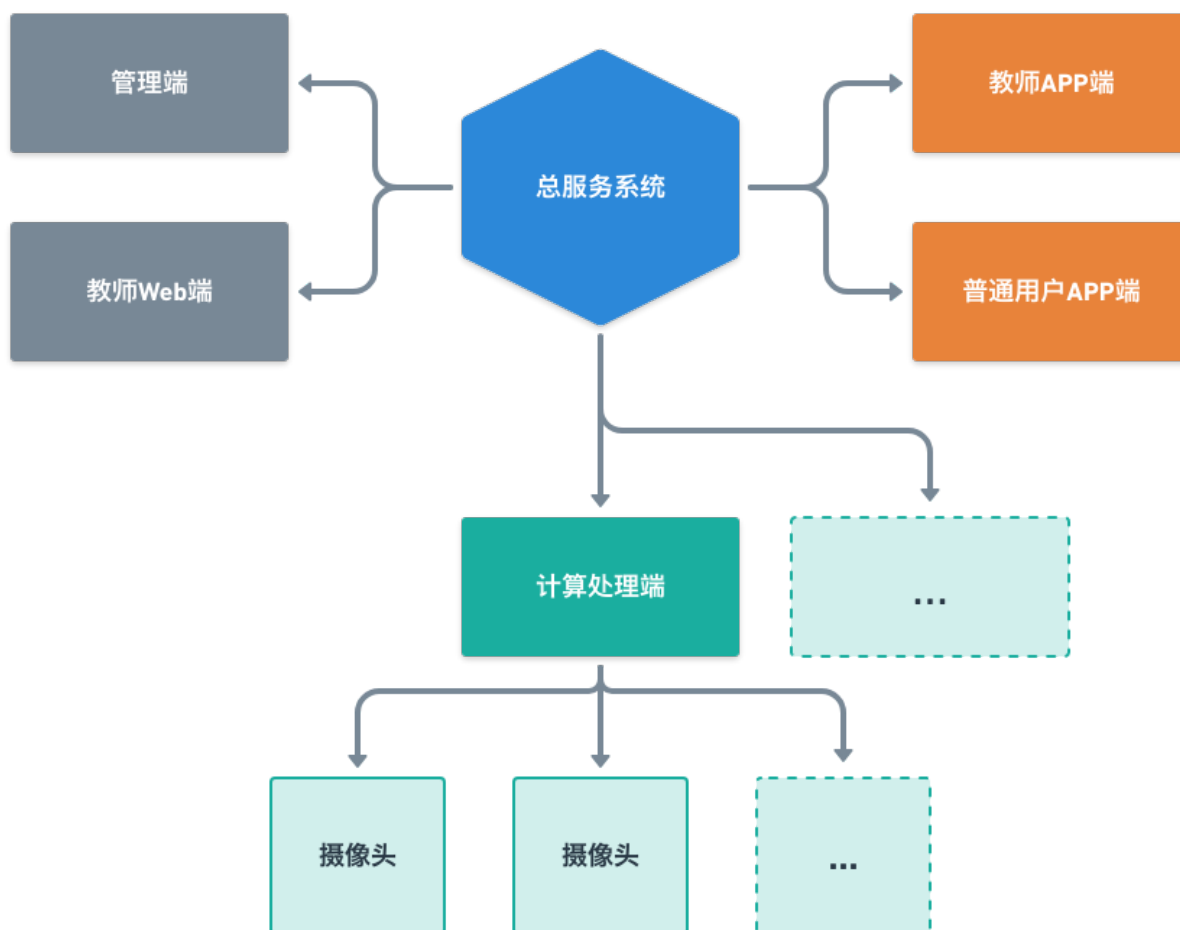
目录

目录	0
系统组成	1
基本介绍	1
功能概述	2
1. 管理端	2
2. 教师端	3
3. 普通用户APP端	4
4. 计算处理端	5
通信过程	6
点名过程	6
上课过程	7
教室和设备状态更新	8
技术栈	9
分析	9
1. 总服务系统	10
2. 管理和教师 Web 端	10
3. 教师 APP 和 普通用户 APP 端	10
4. 计算处理端	10
源码	10
演示	10
管理端	10
仪表盘	11
信息管理	11
信息创建	12
设备状态	13
教室状态	14
教师端	14
课程选择	15
上课情况	17
点名情况	19
演示文稿控制	21
课程记录	22
学生听课状态提醒	23



学生状态记录	24
普通用户APP端	25
课程信息	25
课程记录	26
学生状态记录	27

系统组成

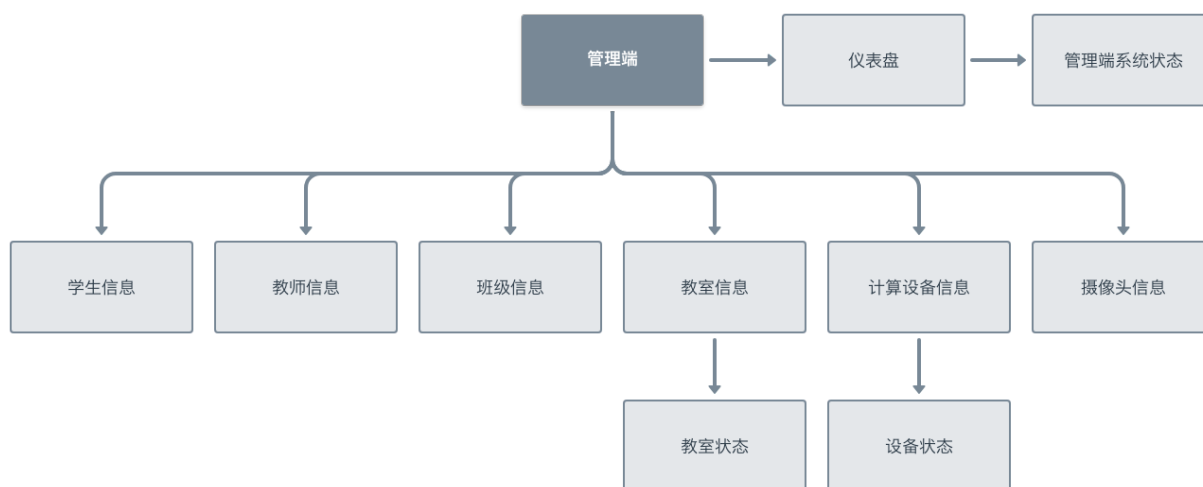


基本介绍

1. 总服务系统：为其它端提供数据 处理、汇总、获取 服务
2. 管理端：系统管理员使用，管理 学生、教师、班级、教室、计算设备、摄像头 数据，系统和设备运行状态、教室状态查看
3. 教师Web端: 教师上课使用，点名、演示文稿功能，配合教师APP
4. 教师APP端：教师使用，提供课程信息、学生信息、课程记录、学生状态记录，配合教师Web端，可以获得点名信息、学生状态信息、学生不专心提醒、演示文稿控制功能
5. 普通用户APP端：学生或家长使用，提供登录用户的 课程信息、用户信息、上课状态记录
6. 计算处理端：嵌入式，配合摄像头用于点名、学生状态、教室状态信息获取，计算处理设备可以有多个，每个计算处理设备可以对应多个摄像头
7. 摄像头：实时画面信息获取，和教室对应

功能概述

1. 管理端



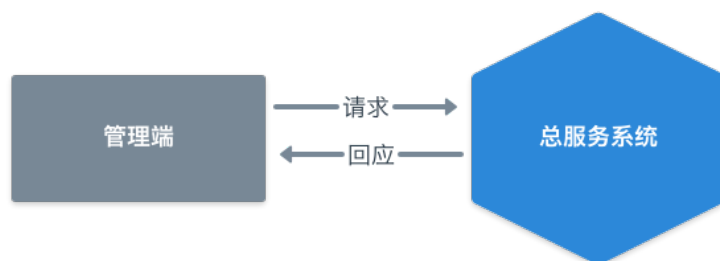
管理端主要是对信息的录入、展示、管理

对信息的录入：[信息创建](#)

对信息的展示：[仪表盘](#)，[设备状态](#)，[教室状态](#)

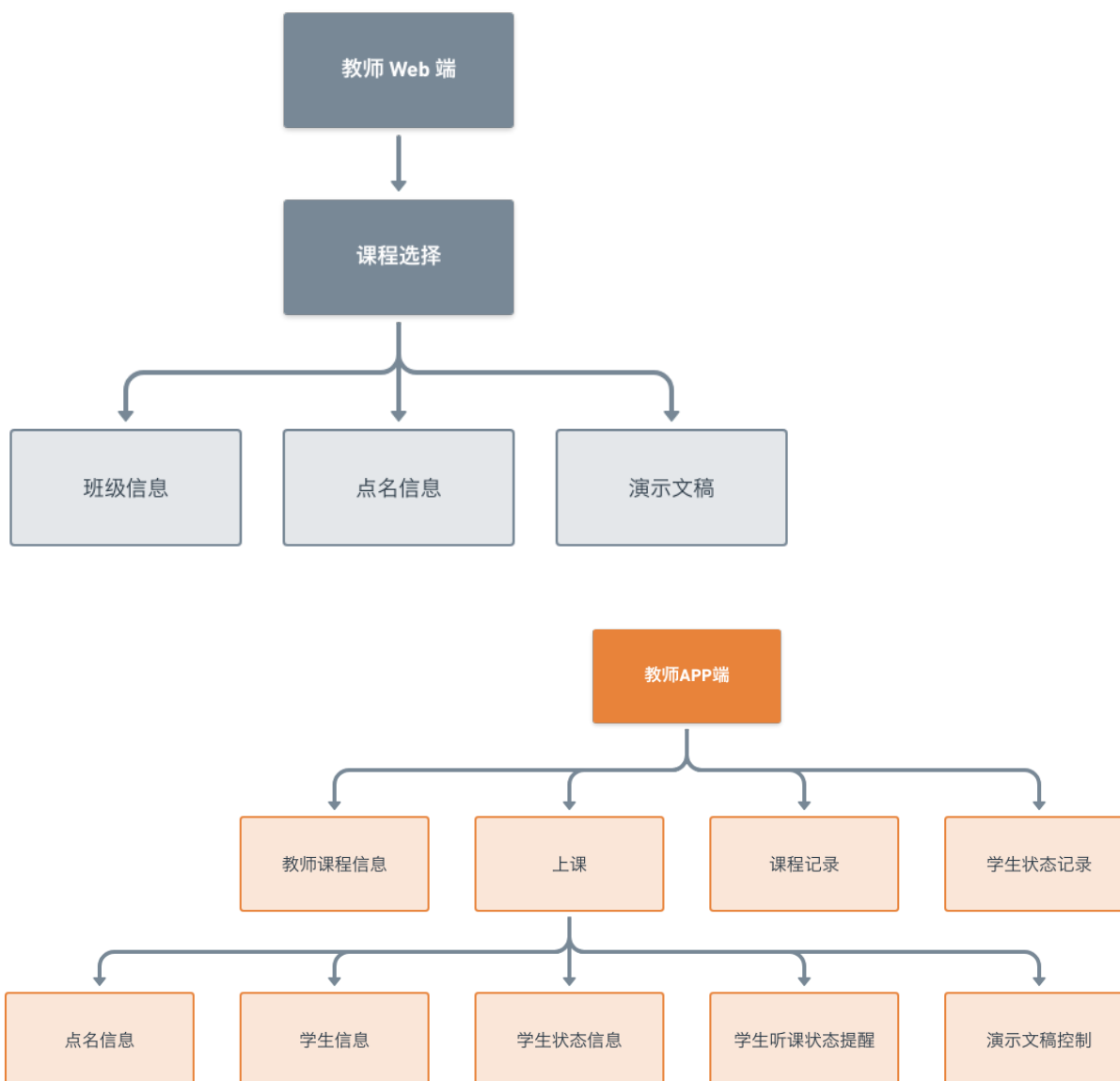
对信息的管理：[信息管理](#)

管理端只与总服务系统通信，且只有普通 HTTP 请求和回应



2. 教师端

教师端分为 Web 和 APP 两块

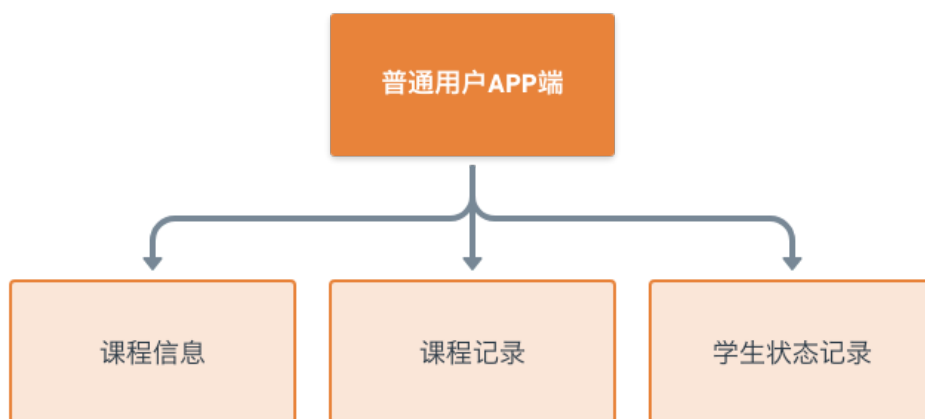


Web 端只在上课时使用，控制课程的开始和结束，Web 端只与总服务系统通信，两种方式：WebSocket 和 HTTP 请求，WebSocket 用于上课时 Web 端与 APP 端的间接实时通信以及点名（[详情](#)），HTTP 请求用于获取课程信息。

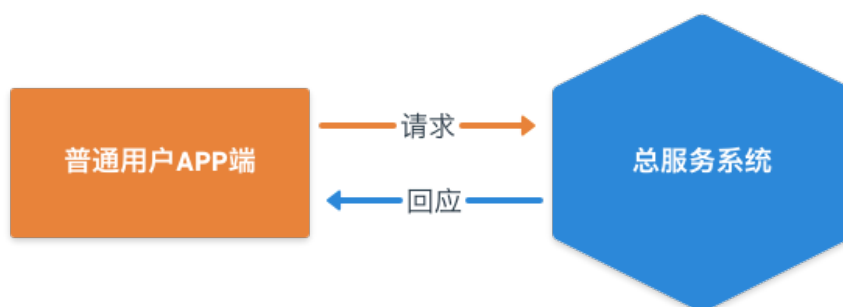
APP 端用于查看课程和学生状态信息，以及对上课的控制，APP 端只与总服务系统通信，和 Web 一样也是两种方式：WebSocket 和 HTTP 请求，WebSocket 用于上课控制，HTTP 请求用于获取课程和学生状态信息。

功能可以概括为：[课程选择](#)，[上课情况](#)，[点名情况](#)，[演示文稿控制](#)，[课程记录](#)，[学生听课状态提醒](#)，[学生状态记录](#)

3. 普通用户APP端

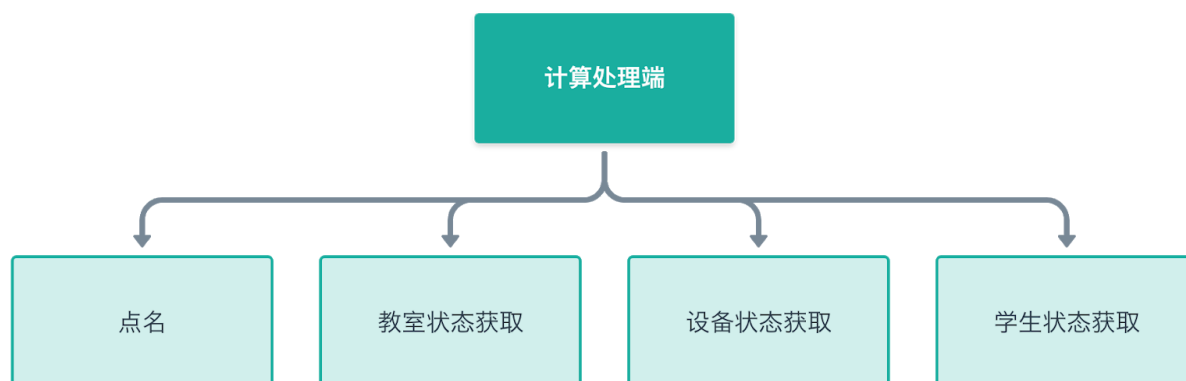


普通用户APP端用于学生查看课程信息、记录，以及自己的学习状态记录，只与总服务系统通信，且只有普通 HTTP 请求和回应。



功能可以概括为：[课程信息](#)、[课程记录](#)、[学生状态记录](#)

4. 计算处理端



点名：获取摄像头图像信息，使用 Face++ API 来得到点名中学生的 FaceToken，再传给总服务系统进行数据比对获得点名信息。

教室状态获取：周期性获取摄像头图像信息，通过 ncnn 下用 mobilenet-ssd 得到教室汇总人的信息，再传给总服务系统记录。

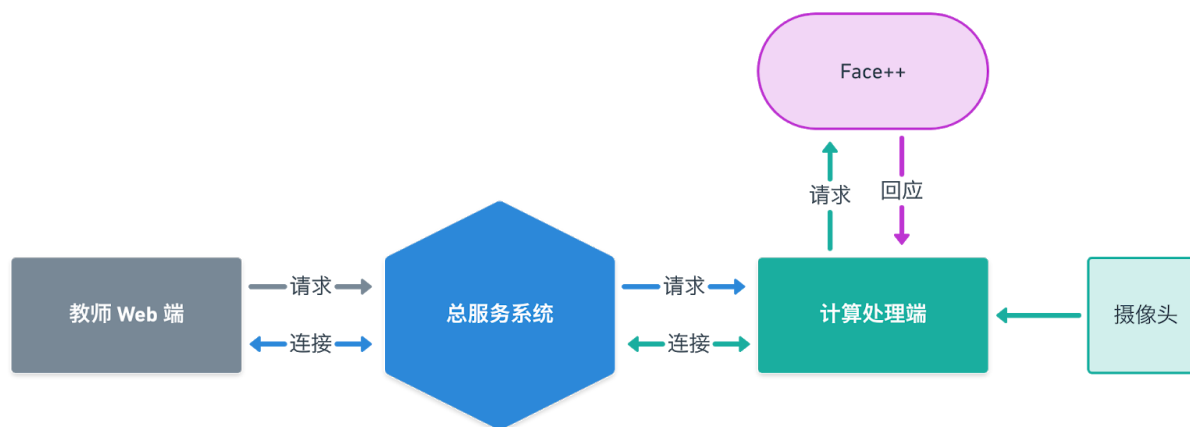
设备状态获取：周期性获取设备运行状态，传给总服务系统记录。

学生状态获取：获取摄像头信息，使用 Face++ API 来获得分析数据，再传给总服务系统进行分析、记录，判断学生听课情况。

计算处理端只与总服务系统通信，教室状态、设备状态、学生状态只有 HTTP 请求，点名使用 WebSocket 与教师 Web 端间接实时通信，但与教师端上课时候有些不同（[详情](#)）

通信过程

点名过程



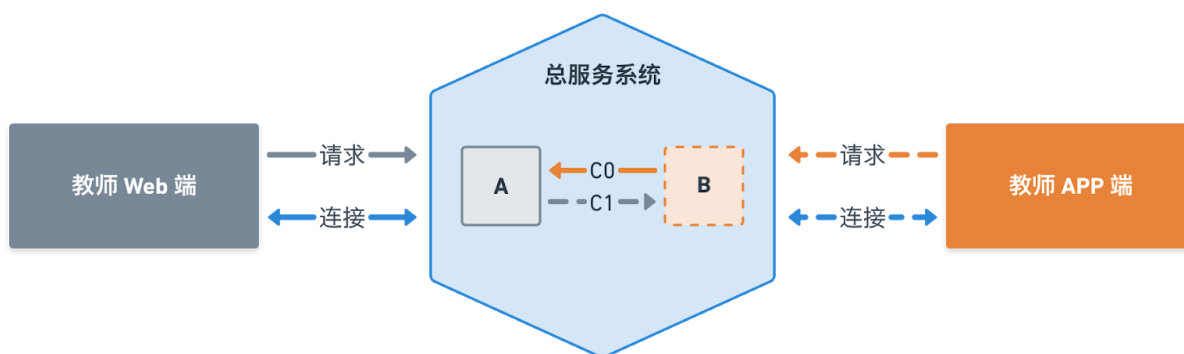
教师 Web 端发起点名请求给总服务系统，总服务系统与教师 Web 端连接 **A** 建立，总服务系统根据课程记录信息找到对应的计算处理端，发起点名请求，计算处理端与总服务系统建立连接 **B**，在总服务系统中，**A** 和 **B** 通过 Golang 中的 Channel 进行通信。

三端连接建立后，计算处理端获取摄像头图像信息，通过 Face++ Detect API 获得每张人脸 FaceToken。

loop { 将 FaceToken 通过 Face++ Search API 获得对应课程中的 FaceToken 传给总服务系统后分析得到对应学生，再传给教师端 **}**

点名完成后，如果存在教师 APP 端的连接，教师 Web 端会将点名信息传给教师 APP 端（具体在下面上课过程中说明）

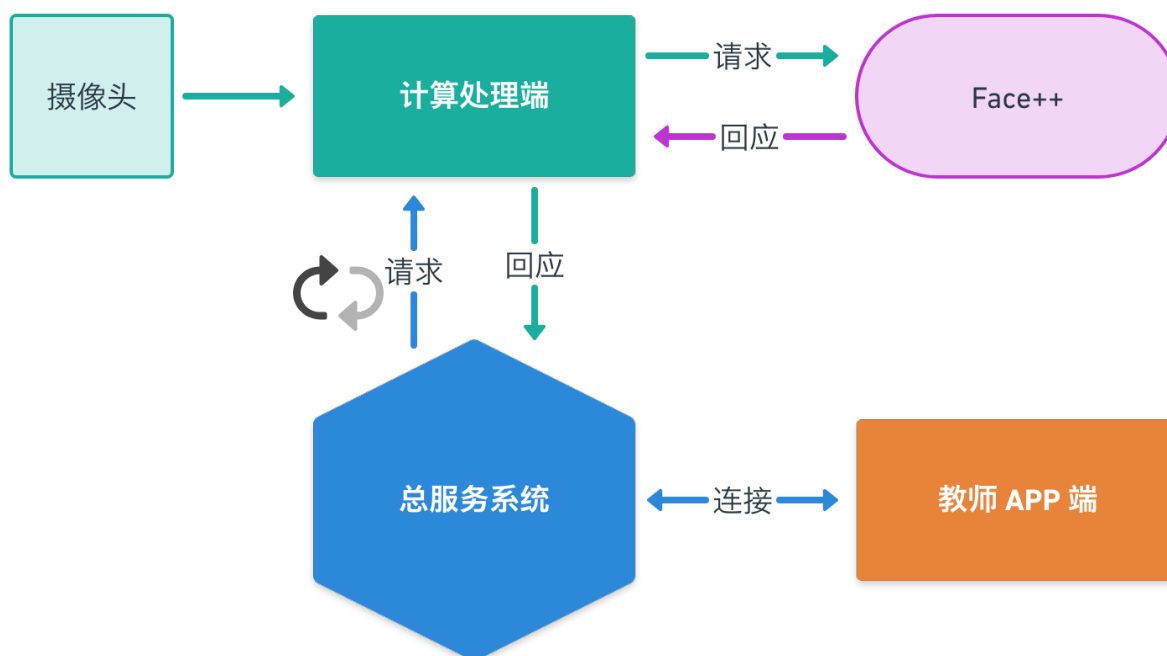
上课过程



教师 Web 端 发起上课请求给 总服务系统，总服务系统与教师 Web 端建立连接 **A**，同时创建 Channel C0，A 只读取 C0 中的数据，用于获得来自教师 APP 端的信息，此时教师 APP 端还未连接，且可以不连接。

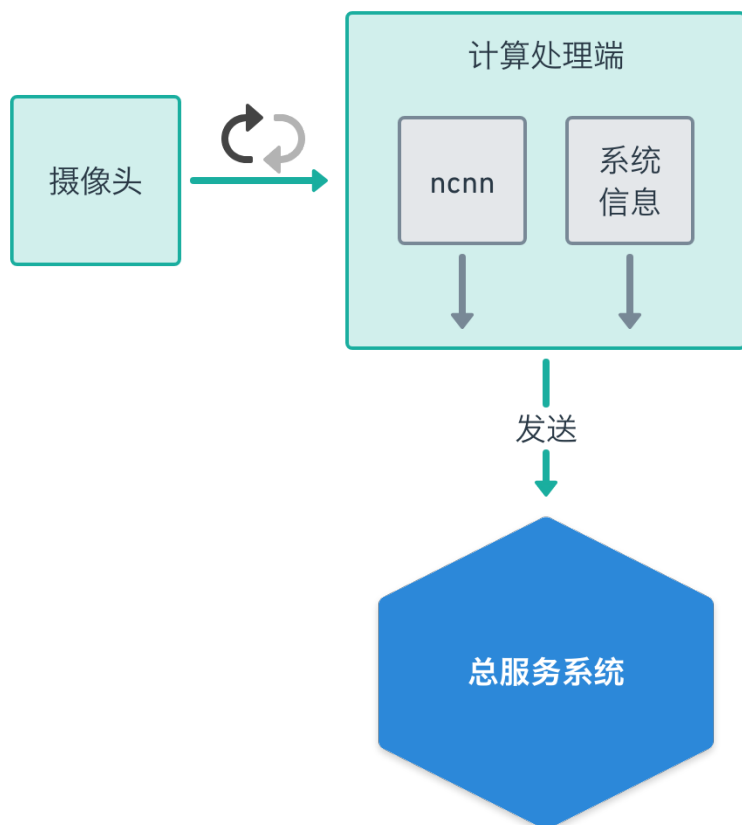
教师 APP 端 可以通过扫描 Web 端上课页面的二维码，或者点击 APP 端课程选择页面的正在上课卡片（如果有上课情况会显示此卡片），进入上课页面（C0 信息取得），此时教师 APP 端与总服务系统连接 **B** 建立，同时创建 Channel C1，用于获取教师 Web 端发送的数据，教师 APP 端通过 C0 告诉连接 **A**，通道 C1 的信息，连接 **A** 发送上课已记录的信息给教师 APP 端，两边连接完全建立。

教师 APP 端 可以随意退出，当 APP 端没有连接的时候，Web 端的数据不会发到 C1，当 C1 已经存在，APP 端再次连接前时就会同时取得 C0 和 C1 的信息，连接后通信恢复。



上课中点名完成后，总服务系统根据点名结果建立基础数据，周期性请求计算处理端的学生状态分析，计算处理端获取摄像头图像信息，请求 Face++ Analysis API 获取数据交给总服务系统进行分析、记录，如果存在学生不认真，传给教师 APP 端进行提醒。

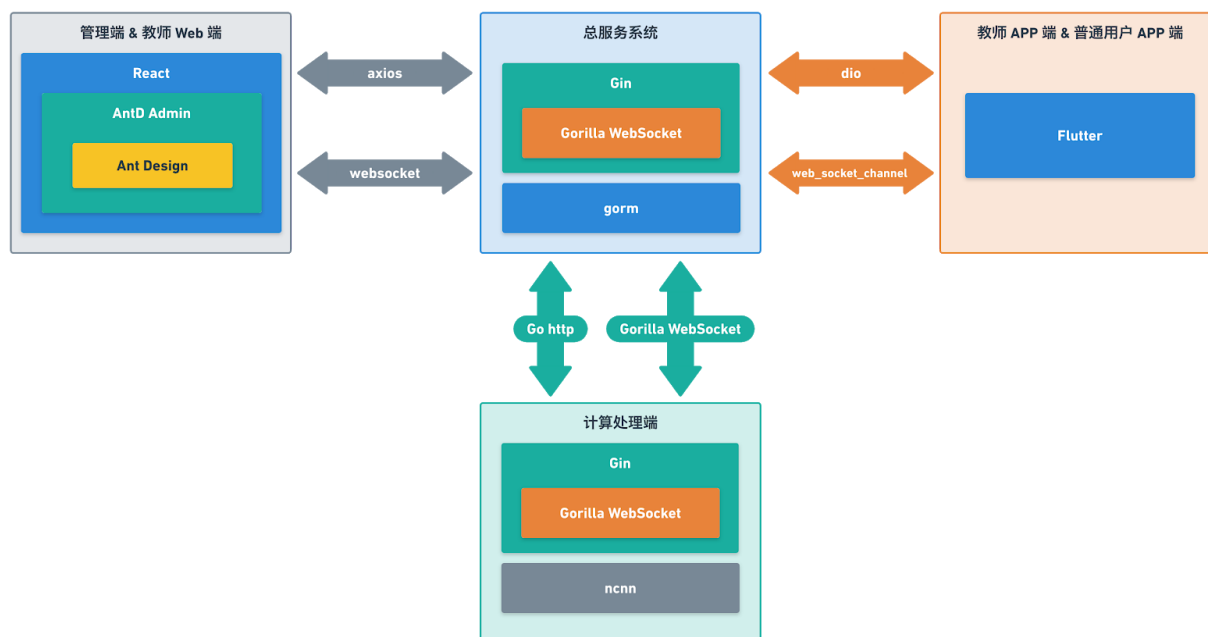
教室和设备状态更新



计算处理端周期性获取摄像头图像和系统状态信息，摄像头图像由 ncnn 下用 mobilenet-ssd 处理后得到教师状态信息，汇总后传给总服务系统。

技术栈

分析



1. 总服务系统

总服务系统由 Gin 框架支持所有的 HTTP 请求，由 Gorilla WebSocket 支持所有的 WebSocket 连接，数据持久化通过 gorm 完全 ORM 化的保存和读取数据。

Gin：由 Go 写成的网络框架，高性能。

Gorilla WebSocket：WebSocket 协议的 Go 实现。

gorm：Golang 下的全功能 ORM 库

2. 管理和教师 Web 端

管理和教师 Web 端以 AntD Admin 脚手架做基础进行开发，UI 组件基本由 Ant Design 支持，由 axios 进行 http 通信，由 html5 websocket 进行 websocket 通信。

React：用于构建用户界面的 JavaScript 库，具有声明式、组件化的特点。

AntD Admin：一套优秀的中后台前端解决方案。

Ant Design：一个 UI 组件库，服务于企业级产品的设计体系，基于确定和自然的设计价值观上的模块化解决方案，让设计者和开发者专注于更好的用户体验。

axios：一个基于 promise 的 HTTP 库。

websocket：WebSocket 对象提供了用于创建和管理 WebSocket 连接，以及可以通过该连接发送和接收数据的 API。

3. 教师 APP 和 普通用户 APP 端

教师 APP 和 普通用户 APP 端通过 Flutter 框架进行开发，由 dio 进行 http 通信，由 web_socket_channel 进行 websocket 通信。

Flutter: Flutter是谷歌的移动UI框架，可以快速在iOS和Android上构建高质量的原生用户界面。Flutter可以与现有的代码一起工作。在全世界，Flutter正在被越来越多的开发者和组织使用，并且Flutter是完全免费、开源的。

dio: dio是一个强大的Dart Http请求库，支持Restful API、FormData、拦截器、请求取消、Cookie管理、文件上传/下载、超时、自定义适配器等。

web socket channel: web_socket_channel包为WebSocket连接提供StreamChannel包装器。它提供了一个跨平台的WebSocketChannel API，该API的跨平台实现，通过底层StreamChannel进行通信，包含dart:io的WebSocket类的实现，以及包装dart:html的类似实现。

4. 计算处理端

计算处理端由 Gin 框架支持所有的 HTTP 请求，由 Gorilla WebSocket 支持所有的 WebSocket 连接，由 GoCV 处理图像数据，由 ncnn 框架处理神经网络计算。

Gin: 由 Go 写成的网络框架，高性能。

Gorilla WebSocket: WebSocket 协议的 Go 实现。

GoCV: 为OpenCV 4计算机视觉库提供Go语言绑定。

ncnn: 一个高性能神经网络前向计算框架。

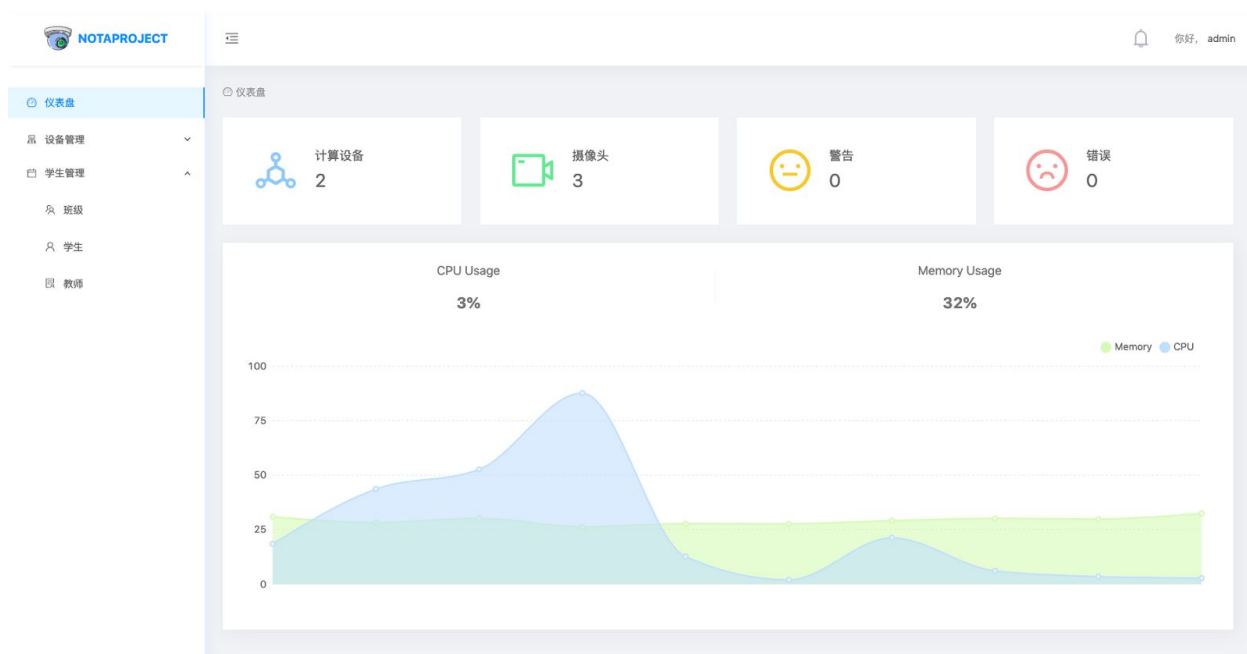
源码

[GitHub](#)

演示

管理端

仪表盘



信息管理

NOTAPROJECT

你好, admin

仪表盘 / 学生管理 / 学生

搜索学号 搜索班级号 搜索 重置 创建

<input type="checkbox"/>	头像	学号	姓名	班级	Face Token	操作
<input type="checkbox"/>		999	Z.J.Y	4	4cafd438d3bf1c15428198c72cb32364	
<input type="checkbox"/>		019	S	2 3	0c0cb45918a6d58d5db1e20b9ec472f1	
<input type="checkbox"/>		018	R	2 3	a6537701fa53d715ebfe599b4b170315	
<input type="checkbox"/>		017	Q	2 3	74e97eb5caac71204074d736d4fa6099	
<input type="checkbox"/>		016	P	2 3	d1c5f11ce396cc918675c61ad5f1a45a	
<input type="checkbox"/>		015	O	2 3	1b58a3ac718aa838bb7be91a3c0b6f2e	
<input type="checkbox"/>		014	N	2 3	159f0fb0905c7ea888dc58a947124ab2	
<input type="checkbox"/>		013	M	2 3	5b79de48e806b627a7760755361c035f	
<input type="checkbox"/>		012	L	2 3	dd6eb7c42f599b6d5acb04aa0cec7c16	
<input type="checkbox"/>		011	K	2 3	050281a0b3bf1d7fe03ac1a92133d1fc	

共 20 条记录 < 1 2 > 10 条/页 跳至 页

信息创建

NOTAPROJECT

你好, admin

仪表盘

设备管理

学生管理

班级

学生

教师

仪表盘 / 学生管理 / 学生

搜索学号

搜索班级号

搜索

重置

创建

	头像	学号	姓名	班级	Face Token	操作
<input type="checkbox"/>		999			5f1c15428198c72cb32364	<div></div>
<input type="checkbox"/>		019			1a6d58d5db1e20b9ec472f1	<div></div>
<input type="checkbox"/>		018			53d715ebfe599b4b170315	<div></div>
<input type="checkbox"/>		017			1ac71204074d736d4fa6099	<div></div>
<input type="checkbox"/>		016			96cc918675c61ad5f1a45a	<div></div>
<input type="checkbox"/>		015			18aa838bb7be91a3c0b6f2e	<div></div>
<input type="checkbox"/>		014			5c7ea888dc56a947124ab2	<div></div>
<input type="checkbox"/>		013			0b78ue4ed08b627a7760755361c035f	<div></div>
<input type="checkbox"/>		012	L	<div>23</div>	dd6eb7c42f599b6d5acb04aa0cec7c16	<div></div>
<input type="checkbox"/>		011	K	<div>23</div>	050281a0b3bf1d7fe03ac1a92133d1fc	<div></div>

共 20 条记录 < 1 2 > 10 条/页 跳至 页

创建学生信息

* 学生姓名: SSS

* 学号: 020

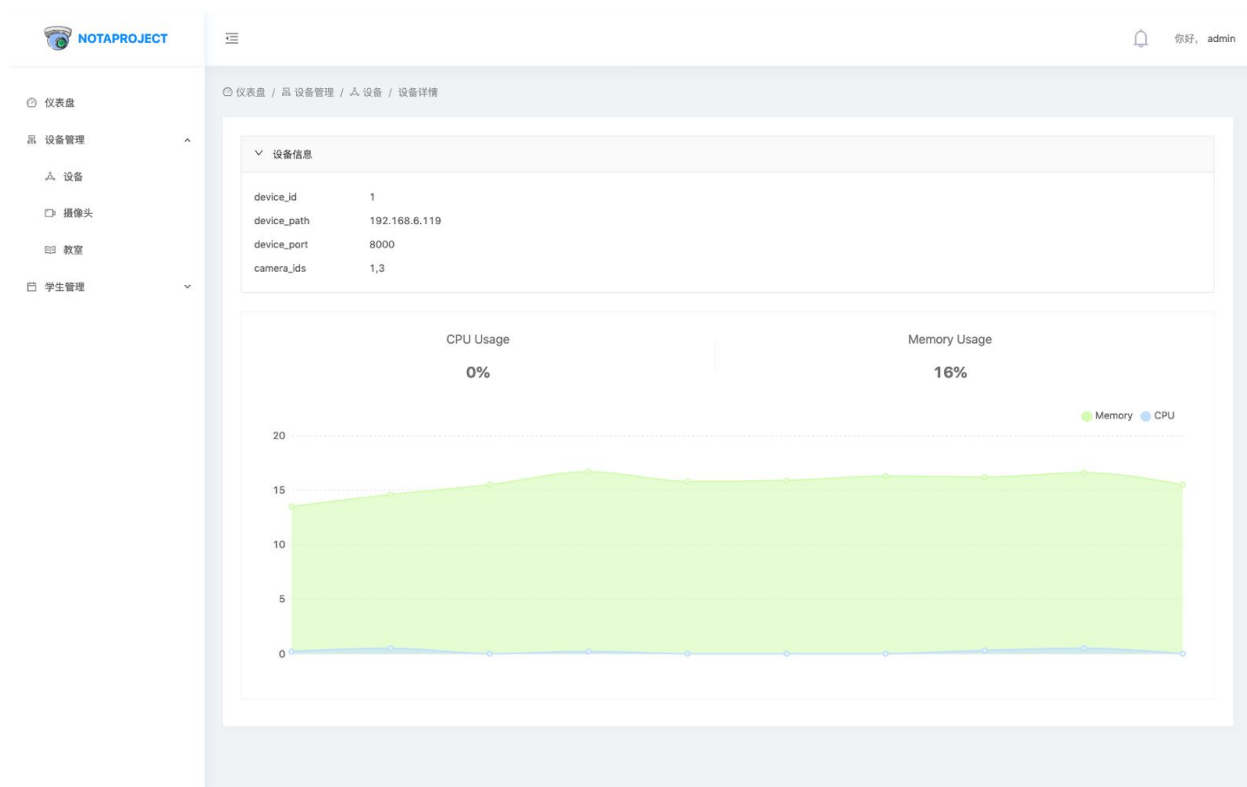
更新密码:

头像图片:

取消

确定

设备状态



教室状态

NOTAPROJECT

你好, admin

仪表盘

设备管理

设备

摄像头

教室

学生管理

仪表盘 / 设备管理 / 教室 / 教室详情

教室信息

classroom_no	1801
camera_id	1

教室状态


更新时间: 2019-02-16 13:58:10 人数: 15



教师端

课程选择

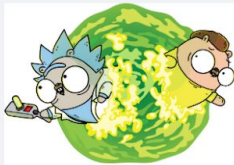


NOTA PROJECT


你好, 001

上课


上课



BBB
ID: 2





LALALA
ID: 3



TEST
ID: 4

上课情况





你好, 001

上课

班级信息

class_id

3

class_name

LALALA

class_image

1548421759044696000.png

face_count

19

faceset_token

2f43bf3efee3ed900fcb47d521ef68f9

classroom_no

1801

student_nos

001,002,003,004,005,006,007,008,009,010,011,012,013,014,015,016,017,018,019

teacher_nos

001

上课

下课

二维码

上传演示文稿

@ 06_Introduction to Computer Vision

点名信息

点名

开始




Introduction to convnets

CS 20: TensorFlow for Deep Learning Research
Lecture 6
1/31/2017

Page 1 of 81

点名情况





你好, 001

上课

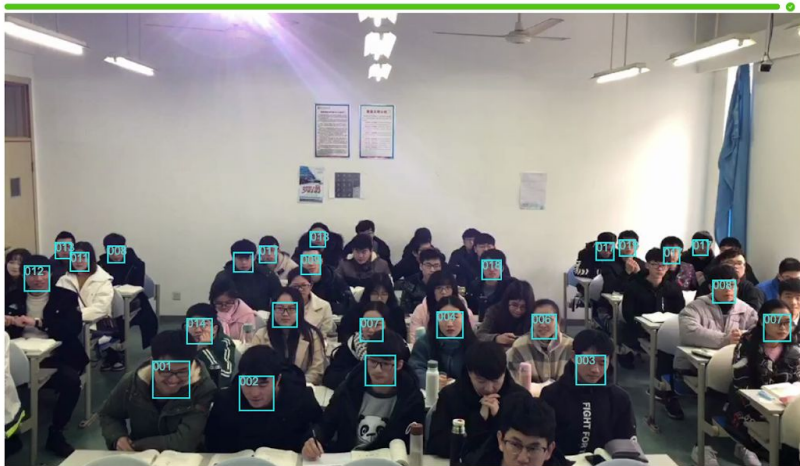
上课 / 上课详情

> 班级信息

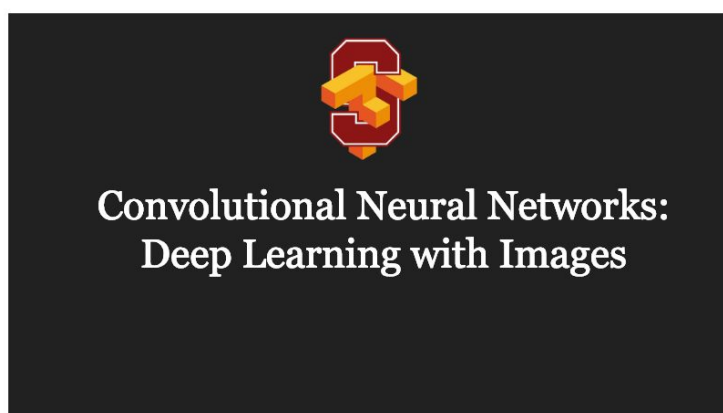
< 点名信息

点名

青藤汁



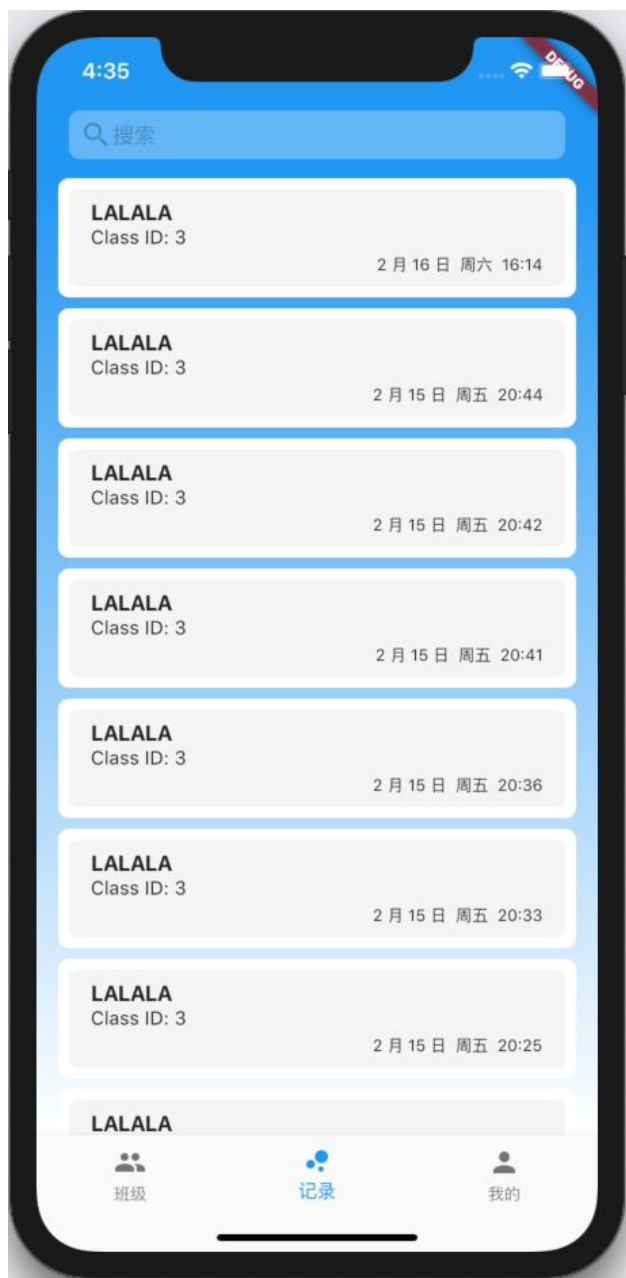
演示文稿控制



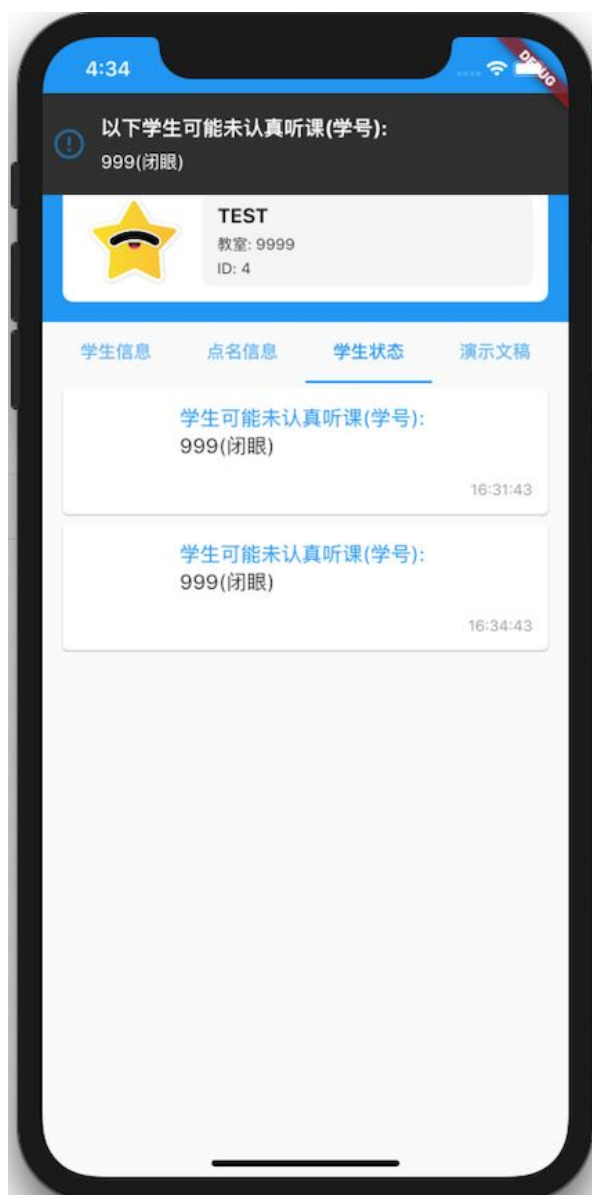
Page 3 of 81



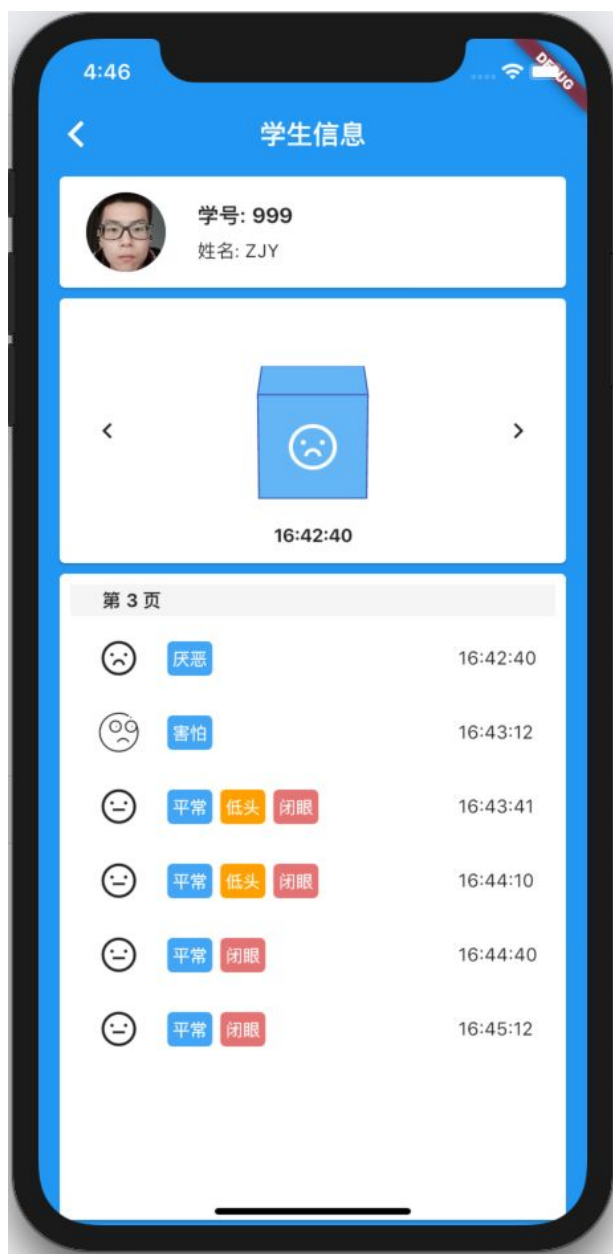
课程记录



学生听课状态提醒

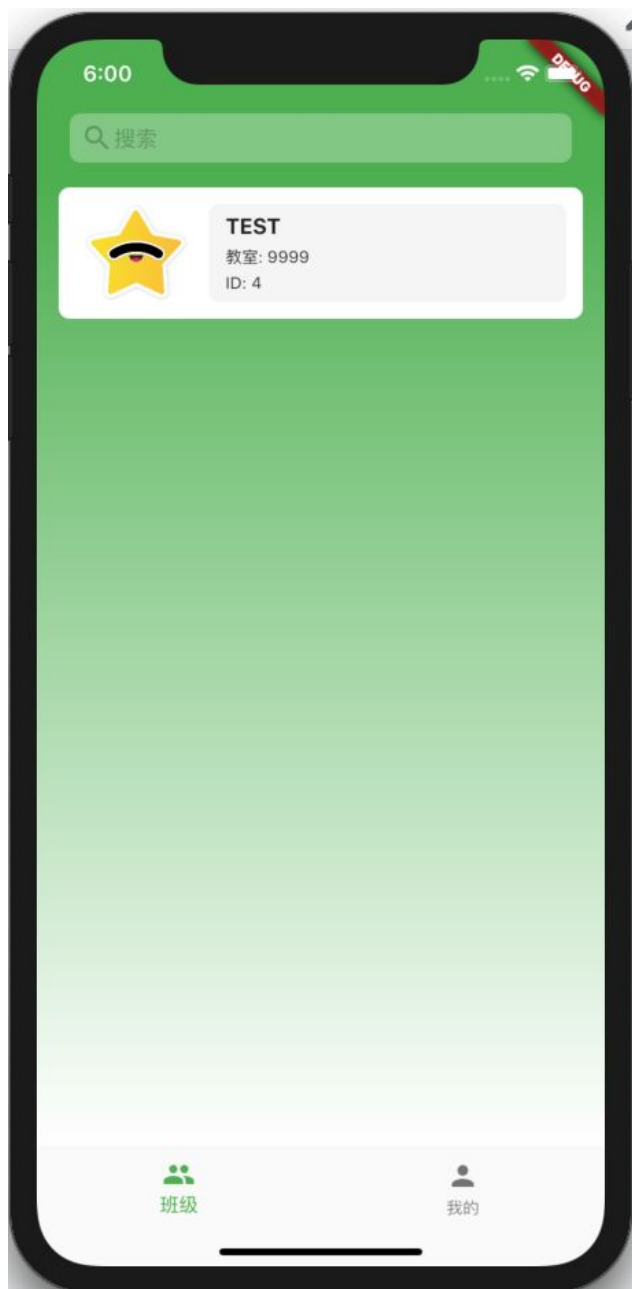


学生状态记录

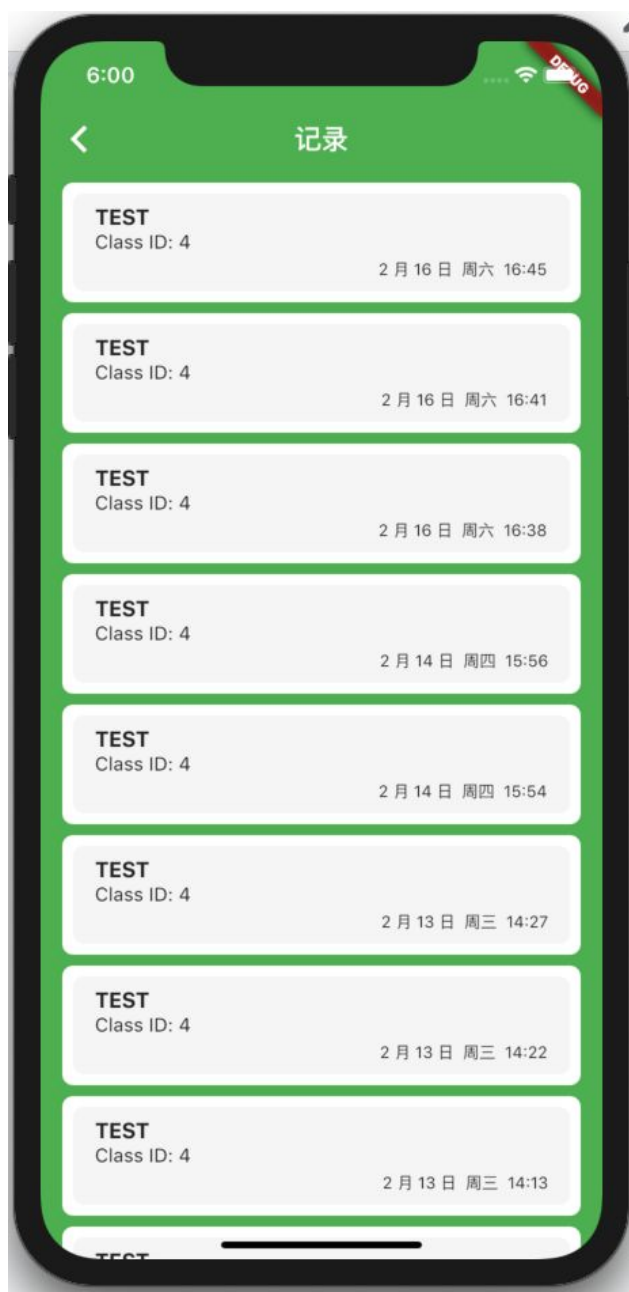


普通用户APP端

课程信息



课程记录



学生状态记录

