

文件编号: **GeeCat-SWC2020-20200047**

受控状态: ☒ 受控 ☐ 非受控

保密级别: ☐ 公司级 ☐ 部门级 ☒ 项目级 ☐ 普通级

采纳标准: **CMMI DEV V1.2**



轻验

AirAuth

项目开发文档

Version V1.0.8

2020.06.05

Written by GeeCat



GeeCat

All Rights Reserved

目录

1 项目概述	1
1.1 项目背景	1
1.2 项目定位	1
1.2.1 应用场景	1
1.2.2 目标人群	1
1.3 项目方案	1
1.4 项目目标	2
1.5 项目价值	2
1.5.1 用户价值	2
1.5.2 行业价值	3
1.5.3 研究领域价值	3
2 开发计划	3
2.1 最终呈现形式	3
2.2 主要功能描述	3
2.2.1 多场景调用接口	3
2.2.2 面部认证功能	3
2.2.3 生物手环认证功能	3
2.2.4 授权管理功能	3
2.2.5 快应用卡片快速入口	4
2.3 运行环境	4
2.4 验收标准	4
2.5 关键问题	4
2.6 进度安排	4
2.7 开发预算	5
3 可行性分析	6
3.1 技术可行性分析	6
3.1.1 功能简介	6
3.1.2 技术要素	6
3.2 资源可行性分析	6
3.3 市场可行性分析	6
3.3.1 与现行软件对比	7
4 需求分析	8
4.1 数据需求	8
4.1.1 静态数据	8
4.1.2 动态数据	8
4.1.3 数据词典	8
4.1.4 数据采集	9
4.2 功能需求	10
4.2.1 功能模块结构图	10
4.2.2 核心功能模块描述	10
4.2.3 核心模块用例规约	11

4.2.4 系统用例图.....	13
4.3 性能需求.....	13
4.3.1 时间特性.....	13
4.3.2 适应性.....	13
4.4 界面需求.....	14
4.4.1 快应用卡片界面.....	14
4.4.2 授权管理界面.....	15
4.4.3 认证服务调用界面.....	15
4.4.4 认证服务功能界面.....	16
4.4.5 认证结果界面.....	16
4.5 接口需求.....	17
4.5.1 硬件接口.....	17
4.5.2 软件接口.....	17
4.6 其他需求.....	17
5 概要设计.....	17
5.1 处理流程.....	17
5.2 总体结构设计.....	18
5.3 功能设计.....	19
5.4 用户界面设计.....	19
5.5 数据结构设计.....	19
5.5.1 逻辑结构设计.....	19
5.5.2 物理结构设计.....	21
5.6 接口设计.....	21
5.6.1 外部接口.....	21
5.6.2 内部接口.....	22
5.7 错误/异常处理设计.....	22
5.7.1 错误/异常输出信息.....	22
5.7.2 错误/异常处理对策.....	23
5.8 系统配置策略.....	23
5.9 系统部署方案.....	23
5.10 其他相关技术与方案.....	23
6 数据库设计.....	24
7 详细设计.....	26
7.1 身份认证功能模块.....	26
7.1.1 功能描述.....	26
7.1.2 性能描述.....	26
7.1.3 输入.....	26
7.1.4 输出.....	26
7.1.5 程序逻辑.....	27
7.1.6 限制条件.....	27
7.2 授权管理功能模块.....	28
7.2.1 功能描述.....	28
7.2.2 性能描述.....	28
7.2.3 输入.....	28

7.2.4 输出.....	28
7.2.5 程序逻辑.....	29
7.2.6 限制条件.....	29

记录更改历史

序号	更改原因	版本	作者	更改日期	备 注
1	初稿	V0.0.0	队员 AB	2019/10/15	——
2	完善需求分析	V0.1.0	队员 AC	2019/10/17	——
3	更新需求分析	V0.1.1	队员 AC	2019/10/19	——
4	编写可行性分析	V0.2.0	队员 AD	2019/10/21	——
5	更新可行性分析	V0.2.1	队员 BCD	2019/10/21	——
6	编写开发计划	V0.3.0	队员 A	2019/10/21	——
7	更新开发计划	V0.3.1	队员 AB	2019/11/3	——
8	修改	V0.3.2	队员 B	2019/11/8	——
9	编写需求分析	V0.4.0	队员 AB	2019/11/10	——
10	完善需求分析	V0.4.1	队员 BD	2019/12/6	——
11	定义各模块接口	V0.5.0	队员 ACD	2019/12/13	——
12	完善需求分析	V0.6.0	队员 D	2019/12/16	——
13	制定测试计划	V0.7.0	队员 A	2019/12/26	——
14	编写数据库设计	V0.8.0	队员 B	2020/2/1	——
15	更改数据库设计	V0.8.1	队员 AB	2020/2/12	——
16	编写概要设计	V0.9.0	队员 ABCD	2020/2/17	——
17	编写详细设计	V1.0.0	队员 ABCD	2020/3/2	——
18	更新	V1.0.1	队员 B	2020/3/12	——
19	更新	V1.0.2	队员 C	2020/3/13	——
20	更新	V1.0.3	队员 C	2020/4/3	——
21	更新	V1.0.4	队员 A	2020/4/20	——
22	更新	V1.0.5	队员 D	2020/4/29	——
23	更新	V1.0.6	队员 C	2020/5/10	——
24	更新	V1.0.7	队员 B	2020/5/20	——
25	更新	V1.0.8	队员 A	2020/6/5	——

1 项目概述

AirAuth 是一款基于生物特征和快应用的身份认证及授权服务。

1.1 项目背景

随着互联网的不断发展，身份认证已经逐渐演化成了一种网络认证服务。据统计，用户每天需要进行十次以上的身份认证，每次认证都十分耗时。认证场景包括且不限于：APP 登录，Web 登录，授权认证，闸机门禁系统等。

网络认证大部分都是基于“账号-密码”的形式，这种形式存在许多问题：

- (1) **隐私问题：**企业更希望获得用户的隐私信息进行订阅推送，宣传等带来巨大利润的行为。据工信部今年 9 月份发布的《工业和信息化部关于电信服务质量的通告》，2019 年二季度，“12321 举报中心”受理用户关于垃圾短信的举报投诉共计 **166025** 件，环比上升 **90.4%**；
- (2) **容易出现密码泄露或遗忘：**人们经常有这样的体验：不经意间浏览一个博客，想要看到完整内容就需要经历繁琐的注册和登录过程；可能在一个网页注册的账户才几个星期没用就产生了遗忘；在不安全的环境登录造成密码泄露，出现盗号，隐私泄露，诈骗等严重后果。

快应用具有多场景入口，和即用即走的优势，可以将快应用作为接入认证服务的入口，提供多场景认证，并且借助微信扫脸等方案基于快应用终端设备 id 和授信用户可以为企业级认证提供可能。所以 **AirAuth 期望基于用户的生物特征生成生物 ID，提供无需账户密码的认证服务。**

1.2 项目定位

1.2.1 应用场景

AirAuth 的应用场景为页面和智能设备的认证授权如：APP 登录，网页登录，安全认证，闸机门禁等认证场景。

1.2.2 目标人群

需要频繁使用认证服务的用户，例如网页登录，企业园区门禁等。

1.3 项目方案

项目整体方案如下图所示：

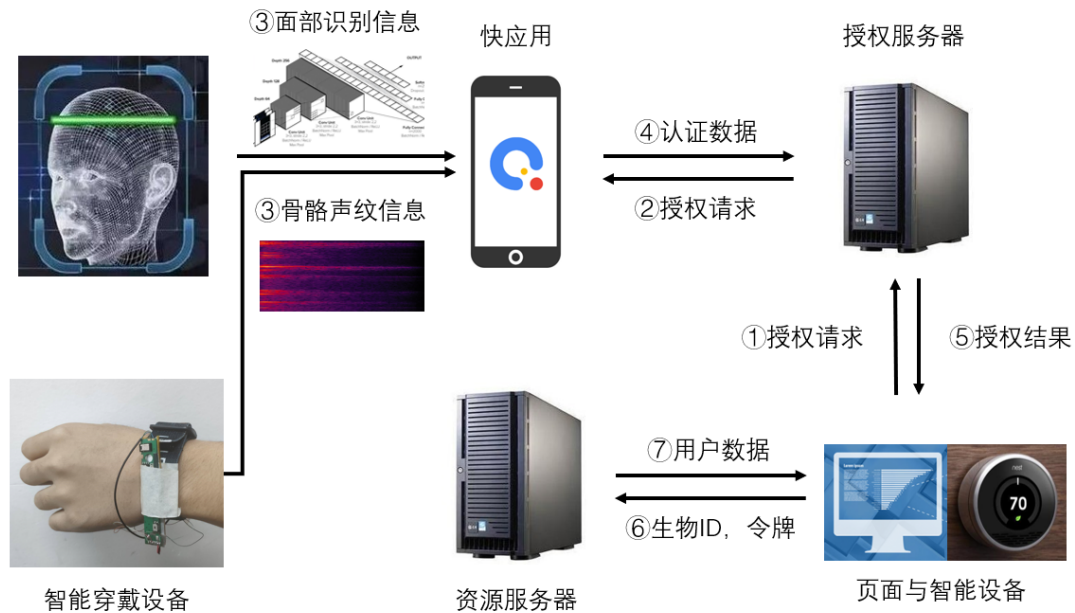


图 1.3.1 项目整体方案

- 1) 当用户在页面或智能设备上尝试登录时，向授权服务器请求授权认证；
- 2) 授权服务器接收请求，唤醒快应用接收用户的身份数据（面部或腕部）；
- 3) 快应用结合 AI 云服务器（自建）对接收的输入数据进行分析处理，推断得到可以表示用户身份的认证数据；
- 4) 授权服务器基于认证数据给予放行；
- 5) 将携带 token 的放行结果返回给页面或智能设备；
- 6) 页面或智能设备通过生物 ID 及令牌向资源服务器请求当前用户数据；
- 7) 资源服务器校验令牌并根据生物 ID 将对应的用户数据返回给页面或智能设备，从而完成授权认证过程。

1.4 项目目标

能够提供可靠生物认证服务，并且基于 OAuth2.0 的通用协议标准和快应用多场景入口及快速调用能力可接入在网页，APP，门禁系统等场景中。

1.5 项目价值

1.5.1 用户价值

AirAuth 提供了无需账号密码，基于快应用终端设备的生物 ID 的快应用能够获取的场景信息和生物信息智能双认证形式为企业级认证提供了可能；解决了传统认证繁琐，暴露隐私等问题，极大优化了用户体验。

1.5.2 行业价值

AirAuth 创新性的利用快应用提供了多场景入口，提供基于 OAuth 的通用授权接口，可以第三方认证（例如：QQ 第三方登录）的服务形式，为软硬件提供身份认证及授权服务；为身份认证和授权服务提供了一种全新的解决方案。

1.5.3 研究领域价值

AirAuth 提出腕部骨骼声纹认证的方式，填补了身份认证和可穿戴领域中，对于可穿戴设备认证的研究的空缺。

2 开发计划

2.1 最终呈现形式

一款以快应用为终端和调用入口，由认证授权服务器支撑，可以通用 OAuth 协议调用的基于生物认证的授权服务。可广泛接入 APP 登录，网页登录，安全认证，闸机门禁等认证场景。

2.2 主要功能描述

2.2.1 多场景调用接口

AirAuth 基于普遍使用的 OAuth2.0 认证授权协议基础上改进。以第三方登录的形式（类似 QQ 第三方登录，微信第三方登录）提供认证授权服务接口；并且同理在智能硬件中也可通过二维码调用。提供链接调用，二维码调用的入口。

2.2.2 面部认证功能

我们提供面部特征作为一种生物认证途径和示例。在用户选择使用人脸认证时，提供基于前置摄像头的人脸识别和采集，以达到识别用户的功能。

2.2.3 生物手环认证功能

在用户佩戴有腕部生物骨纹手环时，AirAuth 可在用户调用认证功能时，选择该方式通过 BLE 读取手环数据。

2.2.4 授权管理功能

AirAuth 提供基于快应用的权限管理功能，授权提醒、展示授权记录、用户可删除被授权方，并且设置认证方式。

2.2.5 快应用卡片快速入口

快应用卡片提供授权记录状态显示和一键直达的快速入口，利用快应用卡片在手机的负一屏展现权限管理的快应用卡片，用户不仅可以快速查看自己的授权状态，还可以直接点击直达管理页面。

2.3 运行环境

快应用服务框架版本 1010+的 Android 手机

服务器端 Linux 系统，安装有 Django2.0 及以上版本及 conda 基础环境

腕部手环采用 esp32 主控，arduino 环境

2.4 验收标准

用户在接入了 AirAuth 认证服务的页面平台或者智能设备平台上，可以通过快应用轻松调用 AirAuth 身份认证服务，并依赖设备 ID 和生物 ID 完成联合查找进行身份认证，并对平台的个性化和隐私服务进行授权。对于生物认证：FRR(拒真率)小于 1%；FAR(误识率)小于 0.05%；ERR(拒登率)小于 1%。

2.5 关键问题

- 1) 参考和改进第三方身份认证及授权协议，来确保认证服务过程的安全可控。
- 2) 以设备绑定可靠用户的形式，缩减生物认证的用户基数，使生物认证可靠性提高。
- 3) 基于骨传导模块采集的骨纹信息提取特征以生成生物 ID。
- 4) 完善智能硬件调用认证的服务接口(蓝牙和二维码)，对智能硬件提供认证服务。

2.6 进度安排

阶段	主要工作	计划起始时间	计划终止时间	责任人
系统需求	调研用户需求	2019/9/30	2019/10/4	队员 ABCD
	论证项目可行性	2019/10/4	2019/10/6	队员 D
	制定项目初步计划	2019/10/6	2019/10/15	队员 A
	确定系统功能	2019/11/0	2019/11/7	队员 B
	确认项目开发计划	2019/11/15	2019/11/20	队员 B
	编写各文档初稿	2019/11/20	2019/12/0	队员 ABCD
概要设计	建立系统总体结构	2020/2/0	2020/2/15	队员 C

	划分功能模块			
	定义各功能模块接口	2020/2/15	2020/3/0	队员 B
	数据库设计	2020/3/1	2020/3/7	队员 D
	制定组装测试计划	2020/3/7	2020/4/10	队员 ABCD
详细设计	设计模块具体实现算法	2020/1/28	2020/3/0	队员 AC
	确定模块间详细接口	2020/2/1	2020/3/0	队员 C
	更新开发文档	2020/3/0	2020/3/7	队员 A
	制定模块测试方案	2020/3/10	2020/3/20	队员 D
	编写测试文档	2020/3/20	2020/4/10	队员 D
系统实现	编写程序源代码	2020/1/28	2020/3/0	队员 AB
	进行模块测试和调试	2020/3/20	2020/4/0	队员 D
系统测试	执行集成测试计划	2020/4/0	2020/4/10	队员 D
	更新测试报告	2020/3/10	2020/4/10	队员 D
结项	完善文档, 准备材料	2020/4/10	2020/4/20	队员 ABCD

2.7 开发预算

开发预算	阿里云轻量应用服务器	57 元
	麦克风模块	130 元
	esp32 主控	90 元
	振子	8 元
	腕带	10 元
	合计:	295 元

3 可行性分析

3.1 技术可行性分析

3.1.1 功能简介

1. 生物认证：通过人脸识别技术和基于腕部骨骼声纹的认证技术，用户可以选择认证方式进行认证，同时根据设备 ID 绑定可信用户缩小认证基数，提高了认证的准确率，和认证的安全性。
2. 基于场景的智能双认证：通过使用地点信息，设备码，可信设备，可信蓝牙等信息，我们可以实现对当前场景的抽象，根据场景的类别，实现对生物特征认证的阈值的智能化控制，从而克服因当前场景受限导致的生物特征认证导致的准确率降低的问题
3. 多场景授权：基于 OAuth2.0 第三方授权协议，可以接入网页，软件，和智能硬件以第三方授权的形式提供授权认证服务，摆脱繁琐操作。
4. 快应用卡片授权管理：通过使用快应用卡片显示授权状态，可作为入口查看授权列表，进行统一管理。

3.1.2 技术要素

1. 深度神经网络
2. 数字信号处理技术
3. 特征工程
4. OAuth2.0 授权协议服务器
5. BLE 协议栈
6. 快应用卡片技术

3.2 资源可行性分析

项目硬件成本极低主要集中在麦克风和主控模块成本。并且采用快应用作为服务载体，主要资源需求在服务器完成协议通讯工作和特征的检索匹配工作。由于我们只是提供认证授权服务，只需部署协议认证授权服务器，无需部署资源服务器，大大降低了服务器的计算能力需求以及硬件消耗。软硬件资源需求低，服务运行期间仅需少量维护人员对服务器进行维护即可，人员资源需求低。

3.3 市场可行性分析

1. 身份认证服务在网页、APP、智能终端上广泛存在，身份认证服务在全球范围内都有着巨大的市场及用户规模。
2. 目前的身份认证服务主要基于账号，并且复杂繁琐、由隐私性问题，在用户体验及各个方面存在很大的提升空间。

3. 需要认证的场景越来越多,除了网站,APP等线上场景,还有园区门禁,安防等线下场景,借助快应用不用下载及多场景入口的优势可以展现极大的易用性。

3.3.1 与现行软件对比

为了对比我们和目前已经应用于互联网服务的身份认证功能,选取设备保存账号密码功能、QQ、Google 身份验证器与我们的身份认证功能的对比绘制如下表格:

	设备保存账号密码	Google 身份验证器	QQ 等第三方登录服务	AirAuth
在互联网中实现第三方登录	√	√	√	√
手机没在身边可验证	×	×	√	√
忘记密码易找回	×	×	×	√
持续、实时认证	×	×	×	√
他人使用设备时可保证安全性	×	×	×	√
能与其他智能设备互联	×	×	√	√
能避免无良商家的数据泄露	×	×	×	√
能实现可移植的服务框架	×	×	×	√
实现跨设备的身份识别	×	×	×	√

Google 身份验证器: 其在用户登录网站时,需要额外输入一次 Google 身份验证器生成的一个 6 位数的动态口令才能验证成功,达到对账号“两步验证”保护的目的。但两步验证的安全性过程繁琐、验证时间长。

设备和浏览器保存账号密码: 可以将账号及密码存储在信任设备及浏览器的云端,免去了每次登陆都要输入账户和密码的繁琐步骤,节省了时间。但是当他人使用设备时可能产生账号与密码的泄露问题。

QQ 等第三方登录服务: 很多网站还可以采用 QQ、微信等第三方账号登录的方式进行身份认证,这种依赖于第三方可信账号的认证模式,本质上仍然与账号密码登录的方式相同,存在账号或密码遗忘时难以找回的问题,且一个账号的密码泄露,会牵连所有关联平台。

4 需求分析

4.1 数据需求

4.1.1 静态数据

包括所有作为控制或参考用的静态数据元素。
如项目配置数据，UI 资源数据等。

4.1.2 动态数据

包括所有在运行中要不断或者在特定的条件下而发生变化的数据，以及在运行中要输入、输出的数据。
如用户录入的信息，文件存储数据，用户所授权的设备和应用信息等

4.1.3 数据词典

用户表：

属性	类型
用户 ID	主键，整型，自动增长
用户昵称	文本
用户腕部骨骼声纹信息	加密文本
用户面部信息	加密文本
用户存储空间配额	整型

应用表：

属性	类型
应用 ID	主键，整型，自动增长
应用名称	文本
接口 ID	文本
接口密钥	加密文本

设备表：

属性	类型
设备 ID	主键，整型，自动增长
设备名称	文本
设备地址	文本

应用授权表：

属性	类型
授权 ID	主键，整型，自动增长

应用 ID	整型, 关联应用表
用户 ID	整型, 关联用户表

设备授权表:

属性	类型
授权 ID	主键, 整型, 自动增长
设备 ID	整型, 关联设备表
用户 ID	整型, 关联用户表

用户授权表:

属性	类型
授权 ID	主键, 整型, 自动增长
被授权用户 ID	整型, 关联用户表
用户 ID	整型, 关联用户表
授权有效期	整型

4.1.4 数据采集

用户腕部骨骼声纹数据:

在快应用通过 BLE 对腕部设备发出采集数据指令, 腕部设备发出 0-1 调制的激励震动并且同时采用麦克风采集 16000 帧 PCM 数据通过蓝牙 notify 分割成 20 字节的循环发送至快应用。

用户面部数据:

用户的面部数据来自于认证应用调用的手机摄像头。当用户腕部骨骼声纹数据不便采集与传输时, 用户可选择采集面部数据来完成认证。通过手机前置摄像头, 录入用户的面部信息。

授权数据:

所有支持本认证服务的第三方应用或网站, 应事先在认证服务器处进行登记, 获取认证服务登录接口 appid 及其密钥。认证服务器将记录应用名称, 对应的接口 ID 和密钥。当用户通过此服务授权给其他应用或智能设备时, 授权的双方信息将被记录, 并上传至服务器。

场景数据:

使用快应用收集用户在认证时的地理位置, 设备码, WIFI 热点 MAC 地址列表, 蓝牙 MAC 地址列表, 上传到服务器中, 并作为模糊聚类算法的数据点, 实现基于场景的智能认证

4.2 功能需求

4.2.1 功能模块结构图

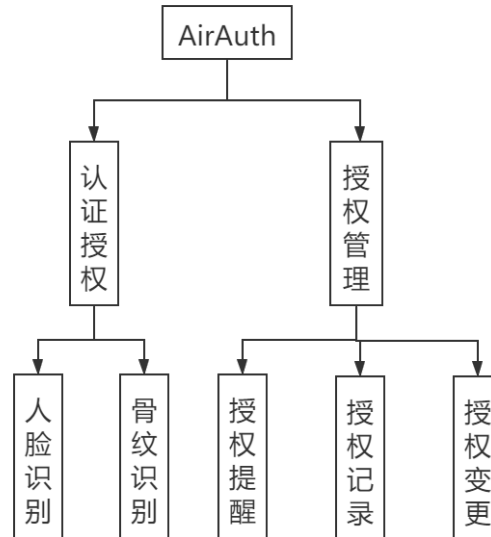


图 4.2.1.1 功能模块结构图

4.2.2 核心功能模块描述

功能模块	功能	功能描述	优先级
身份认证	腕部骨骼声纹认证	可穿戴设备采集用户通过做出特定动作，例如握拳或打响指，产生的独特生物声纹信息，并传输给手机进行认证是否为本人。	高
	面部识别认证	认证应用调用手机前置摄像头获取用户面部信息，进行面部识别，认证是否为本人。	低
授权服务	第三方登录	通过此服务登录其他应用或网站而无需在相应网站注册。	高
	智能设备授权	通过此服务在智能设备处登记授权，从而有权对其进行使用。	高
	用户间授权	提供用户之间相互授权的功能，并可指定授权有效期限。	低
授权管理	取消授权	取消对某个应用、智能设备或用户的授权。	高
	分类查看	将已授权对象的信息，按应用、智能设	低

		备和用户三种类别进行整理，供用户查看和管理。	
	授权详情	可查看授权对象的详细信息。	低

核心功能模块描述

4.2.3 核心模块用例规约

用例名称	腕部骨骼声纹认证
功能简述	可穿戴设备收集用户独特的腕部骨骼声纹信息，与用户事先录入的信息进行比对匹配，验证用户是否为合法用户。
用例编号	001
执行者	用户终端
前置条件	用户完成首次登陆并录入个人腕部骨骼声纹信息，同时正确佩戴可穿戴设备
后置条件	根据认证结果决定用户是否有权访问数据或授权
涉众利益	用户：希望系统准确快速地完成认证
基本路径	①用户做出指定动作 ②可穿戴设备传感器收集用户产生的声纹信息 ③可穿戴设备与用户手机建立蓝牙连接，将数据传至手机 ④手机将数据上传至服务器进行匹配认证 ⑤服务器返回认证结果
扩展路径	声纹信息匹配不符，返回拒绝认证的结果
字段列表	无
设计规则	无
未解决的问题	准确快速的声纹匹配算法
备注	无

腕部骨骼声纹认证用例规约

用例名称	智能设备授权
功能简述	授权功能之一，将用户账号信息在其他智能设备处登记，完成对智能设备的授权。
用例编号	002
执行者	用户终端
前置条件	用户手机与被授权智能设备均开启蓝牙功能
后置条件	用户对智能设备进行授权，有权与其进行交互

涉众利益	用户：希望手机端方便快捷地授权和管理智能设备
基本路径	①用户选择待授权智能设备 ②用户客户端向用户发起认证请求 ③用户通过腕部骨骼声纹认证或面部识别通过认证 ④用户手机与待授权设备建立蓝牙连接 ⑤被授权设备登记用户账号信息
扩展路径	认证过程失败，无法完成授权
字段列表	无
设计规则	无
未解决的问题	智能设备的用户数据存储方案
备注	无

智能设备授权用例规约

用例名称	取消授权
功能简述	用户收回对第三方应用、智能设备或其他用户的授权
用例编号	003
执行者	用户
前置条件	用户进入授权管理界面，查看授权详情
后置条件	用户成功取消对相应对象的授权，并将其从已授权列表中移除
涉众利益	用户：希望取消授权额的对象彻底失去授权，无任何数据保留
基本路径	①用户选定取消授权对象 ②客户端上传取消授权请求 ③服务器删除相应对象的授权信息 ④服务器与相应对象服务器进行交互，删除用户数据 ⑤服务器返回结果
扩展路径	无
字段列表	无
设计规则	无
未解决的问题	是否能够有效地删除授权对象中的用户数据
备注	无

取消授权用例规约

4.2.4 系统用例图

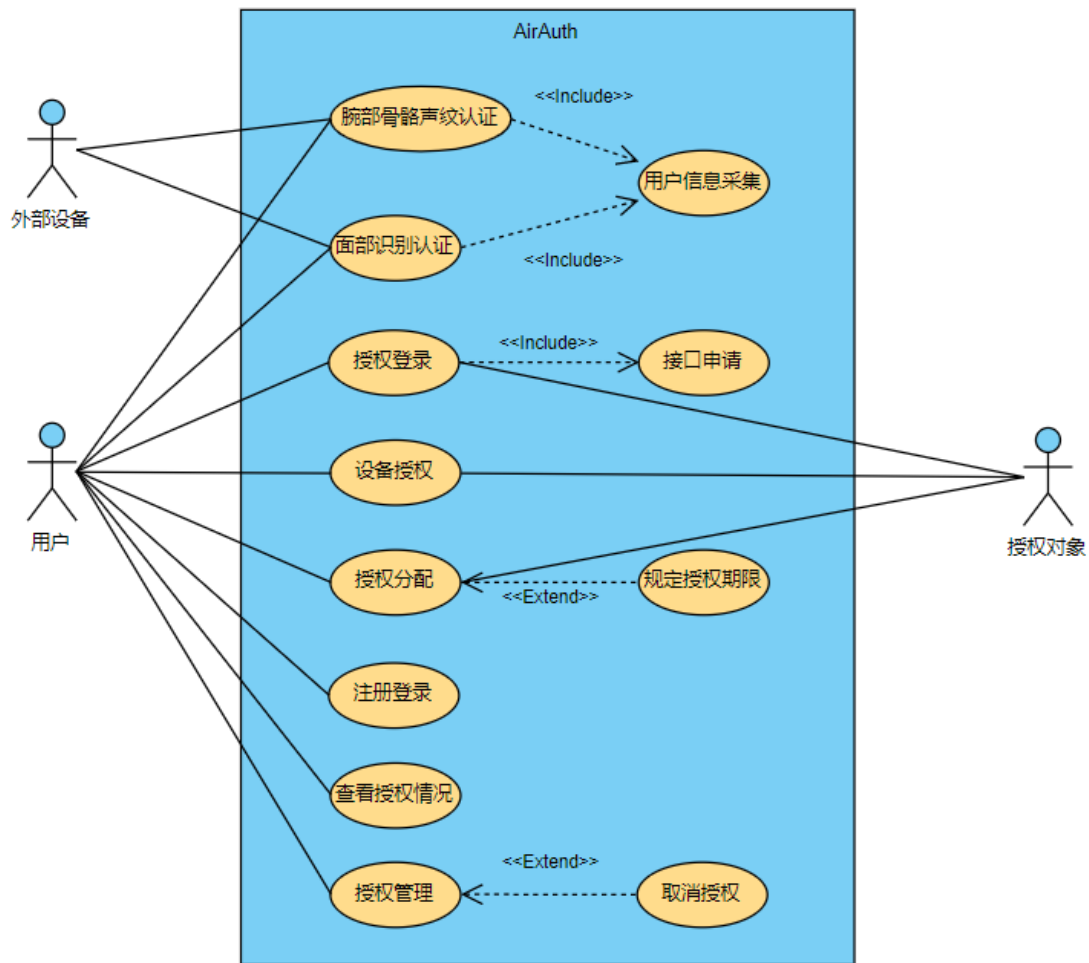


图 4.2.4.1 系统用例图

4.3 性能需求

4.3.1 时间特性

系统应具有快速响应的特性,用户通过外部设备进行认证的平均响应时间应低于 2 秒。用户进行授权操作的数据处理时间应低于 2 秒。(响应)

4.3.2 适应性

Server 部署在 linux 服务器上,在服务器环境正常运行的情况下,快应用终端能在支持快应用的 Android 手机上稳定运行。快应用端出现问题时会将错误信息上报服务器记录在 quickapp.log 文件中,服务器运行出现异常时会将异常信息记录在 server.log 文件中。

4.4 界面需求

4.4.1 快应用卡片界面



图 4.4.1.1 快应用卡片界面

4.4.2 授权管理界面

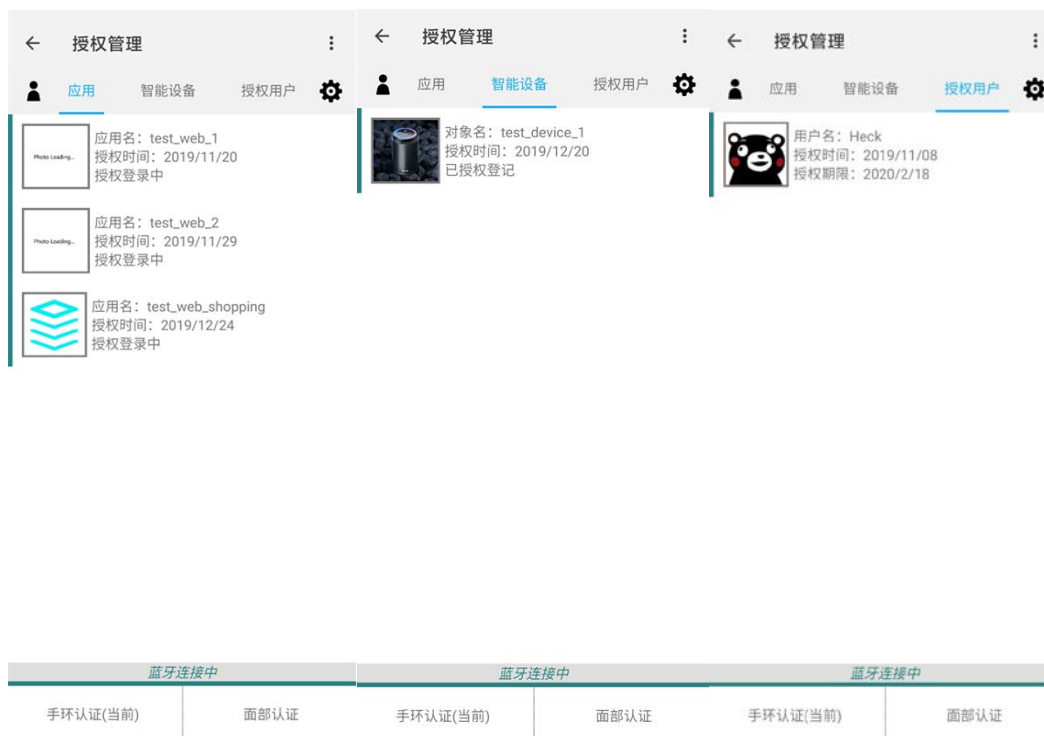


图 4.4.2.1 授权管理界面（应用、智能设备、用户授权）

4.4.3 认证服务调用界面

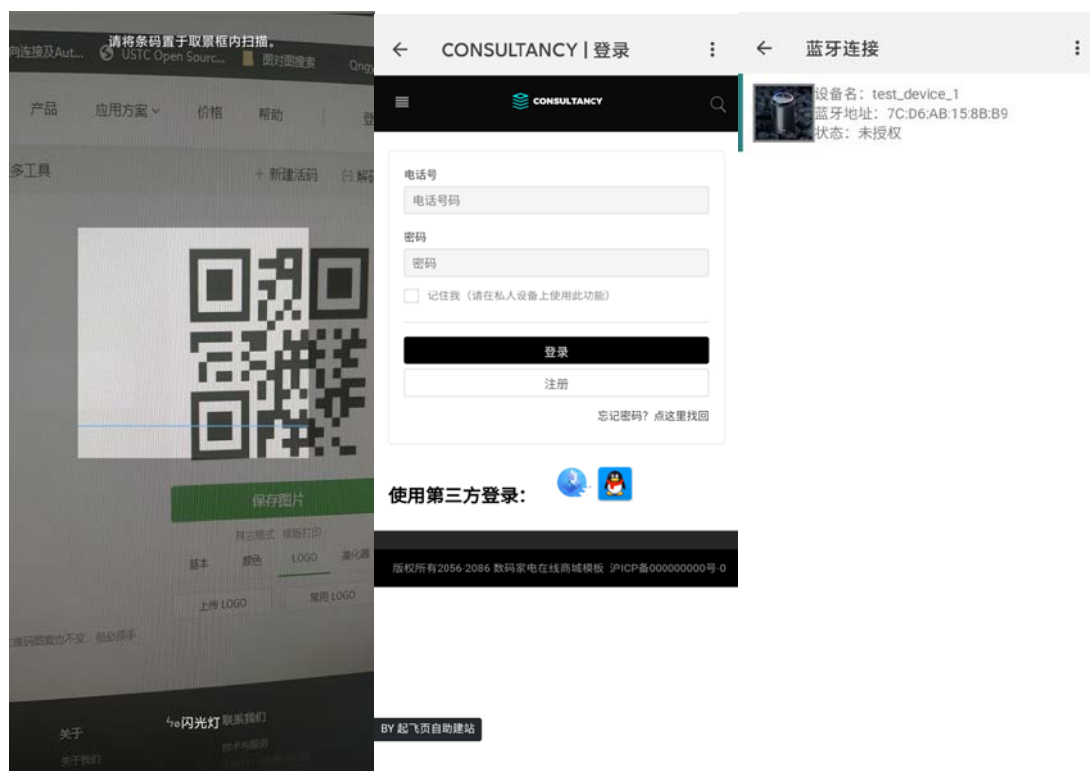


图 4.4.3.1 认证调用接口界面（二维码、手机端第三方登录、智能设备蓝牙调用）

4.4.4 认证服务功能界面

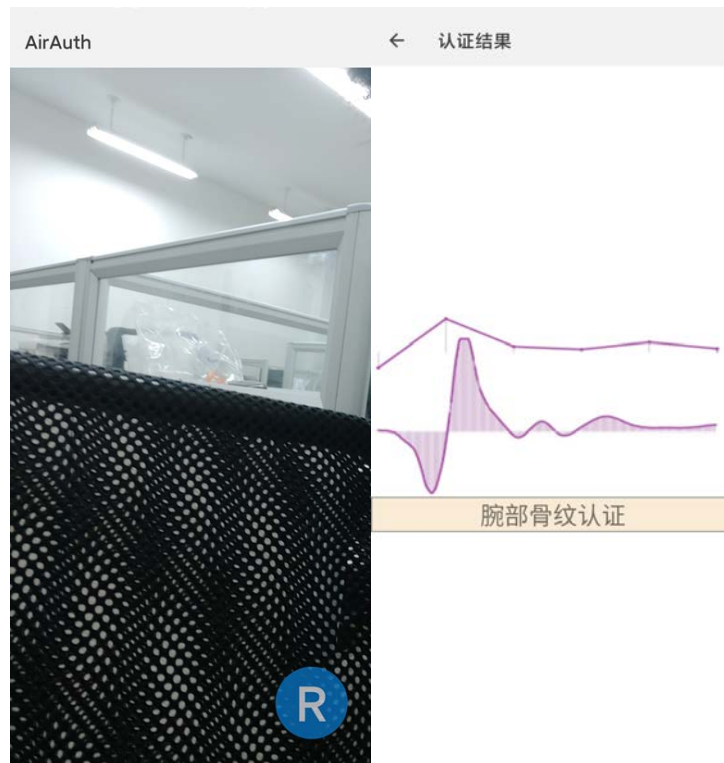


图 4.4.4.1 认证服务功能界面（面部认证、腕部骨骼声纹认证）

4.4.5 认证结果界面



图 4.4.5.1 认证结果界面

4.5 接口需求

4.5.1 硬件接口

若对智能设备进行授权，则需要相应智能设备具有记录、识别和管理授权用户记录的网络后台。

4.5.2 软件接口

若对第三方应用进行授权，则需要第三方应用事先在本项目服务器处进行登记，获取appid和接口密钥。当有用户通过本项目在第三方应用进行登录时，第三方应用将所申请的ID和密钥发送至项目服务器进行验证。

4.6 其他需求

无

5 概要设计

5.1 处理流程

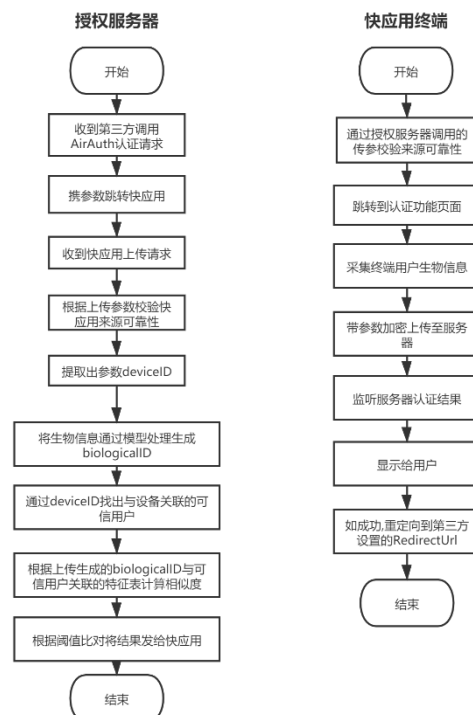


图 5.1.1 逻辑流程图

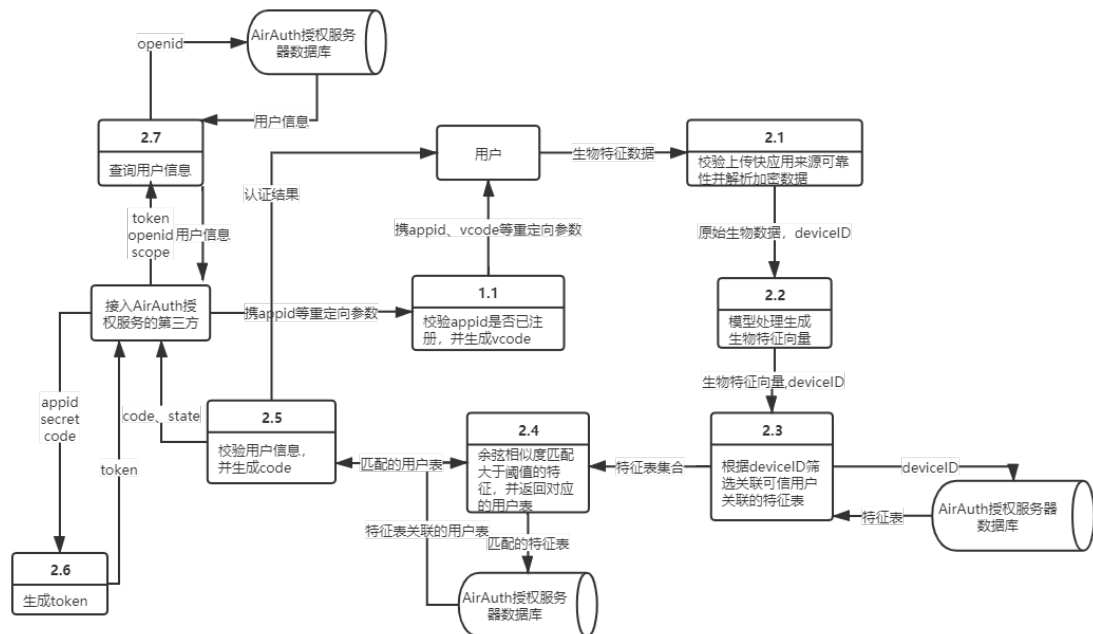


图 5.1.2 数据流图

5.2 总体结构设计

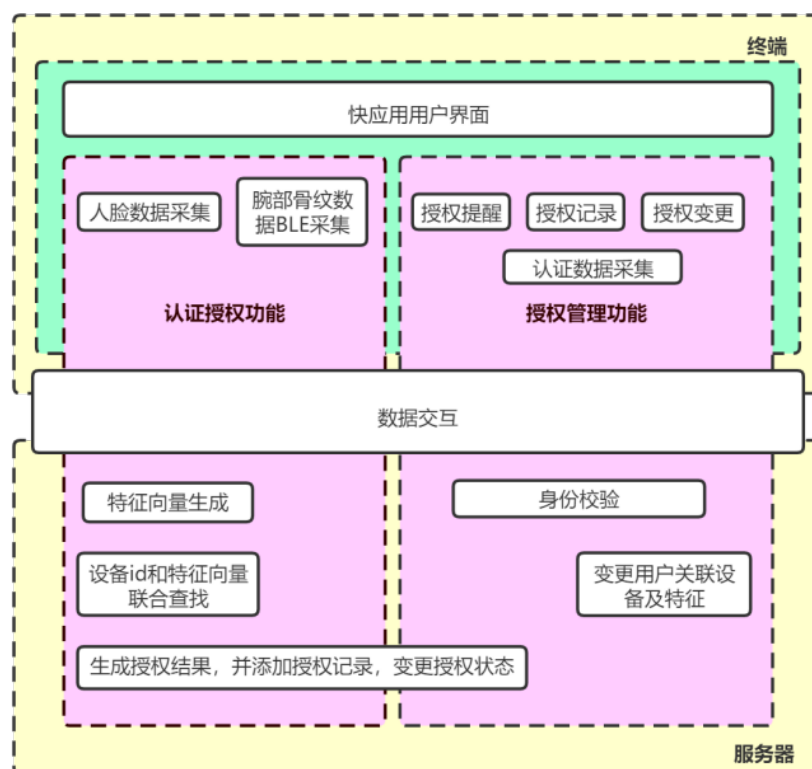


图 5.2.1 模块架构图

5.3 功能设计

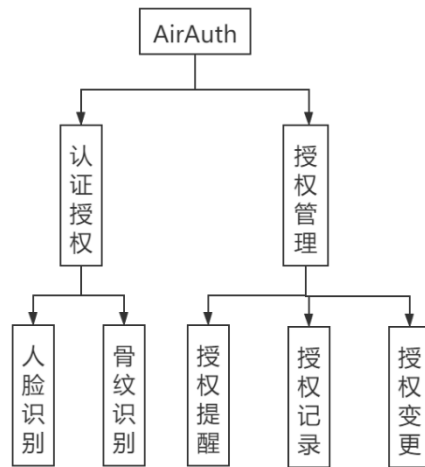


图 5.3.1 功能结构图

5.4 用户界面设计

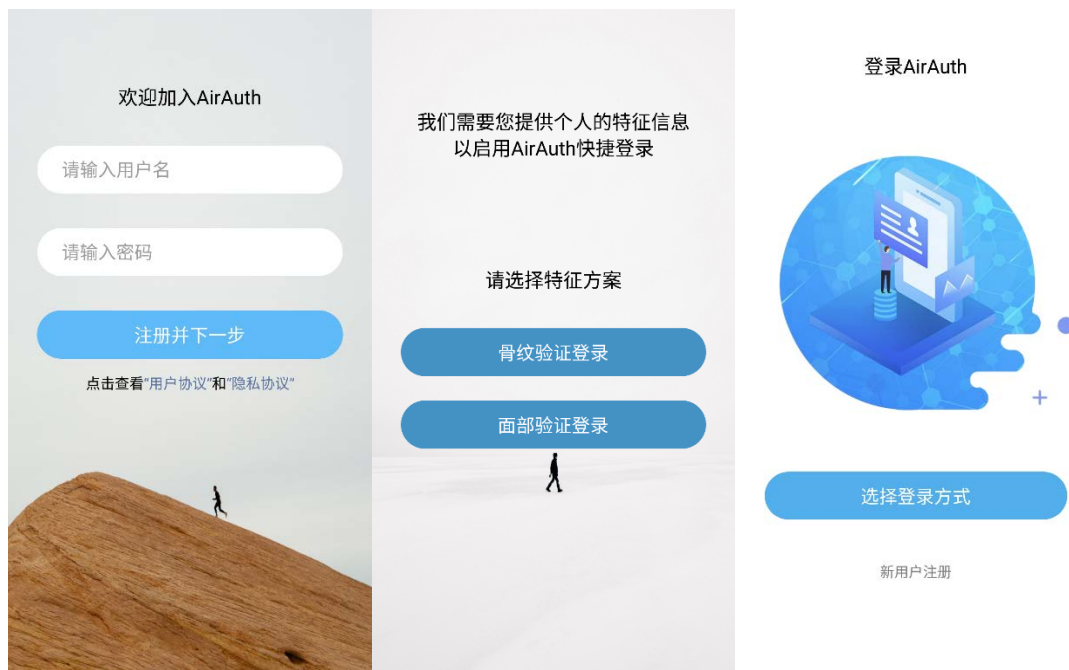


图 5.4.1 用户界面设计图

5.5 数据结构设计

5.5.1 逻辑结构设计

实体：用户 user

属性	解析
user_id	用于记录识别用户身份独一无二的属性，自增，作为主键
user_name	用于记录用户用户名
open_id	用于记录用户对外公开的唯一标识
user_image	用于记录用户头像

实体：第三方网站 third_party_website

属性	解析
client_id	用于记录识别申请授权登录接口的网站独一无二的属性，作为主键
client_key	用于记录 client_id 对应的密钥，其用于验证请求授权的合法性
redirect_uri	用于记录成功授权后网站的回调地址
client_name	用于记录第三方网站名

实体：用户设备 device

属性	解析
device_id	用于记录识别申请用户登录设备独一无二的属性，作为主键
user_id	用于记录 device_id 对应的 user_id

实体：面部特征 face

属性	解析
count	用于记录数据 id，自增，作为主键
user_id	用于记录 face_feature 对应的 user_id
face_feature	用于记录面部特征数据

实体：骨纹特征 bone

属性	解析
count	用于记录数据 id，自增，作为主键
user_id	用于记录 bone_feature 对应的 user_id
bone_feature	用于记录骨纹特征数据

关系：授权 authorize

属性	解析
count	用于记录数据 id，自增，作为主键
type	用于记录授权对象类型
client_id	用于记录被授权对象的 client_id
user_id	用于记录授权用户的 user_id
time	用于记录授权时间

5.5.2 物理结构设计

文件系统

由于训练好的特征提取模型需要在服务运行的时候载入,并且不便于使用数据库进行存储,所以使用文件系统将模型文件存储在服务器上,并且在服务刚开始进行的时候就读取模型,不需要频繁的访问文件系统。

MySQL

由于本项目所需要存储的用户数据都是结构化数据,并且都需要频繁访问,所以使用关系型数据库,并且目前项目的数据总量还比较小,使用速度快,开发简单的 MySQL 比较合适

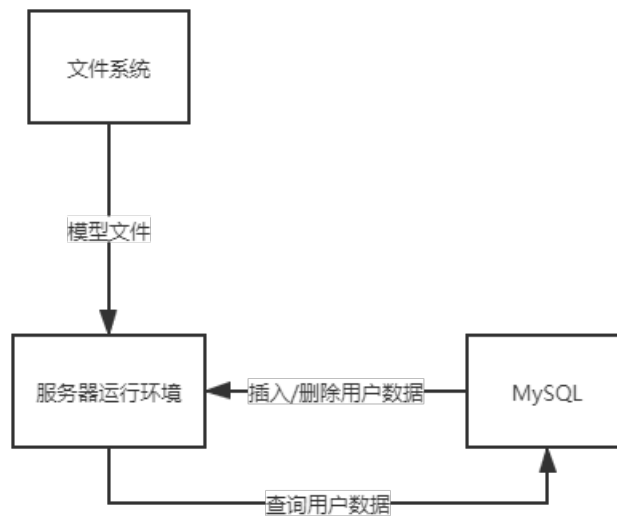


图 5.5.2.1 物理结构设计

5.6 接口设计

5.6.1 外部接口

AirAuth 提供授权登录接口,该接口用于向其他网站提供 AirAuth 第三方授权登录功能。欲使用此接口,其他网站应事先在 AirAuth 申请,注册 redirect_uri 等信息,以获得 client_id 和 client_key。

请求参数

参数	是否必选	类型	说明
client_id	是	string	申请接口时获得的 client_id, 用于识别不同的调用者
redirect_uri	是	string	申请接口时注册的回调地址, 需要将 url 进行编码
client_key	是	string	申请接口时获得的 client_key, 用于验证调用者合法性
state	是	string	随机产生的一段字符串, 用于防止 CSRF 攻击
scope	否	string	表示权限范围

返回参数

字段	类型	说明
code	string	用户授权成功后获得的 Authorization Code
access_token	string	访问令牌
expires_in	uint t	访问令牌有效期
open_id	string	授权用户公开的标识符
state	string	随机产生的一段字符串, 与请求时的 state 一致

5.6.2 内部接口

说明	接口	参数	调用成功返回
加密/解密	aes	action: 指定加密/解密; text: 待加密/解密内容; key: base64 编码形式密钥; option: 算法填充项	text: 经加密/解密生成的文本内容。
获取蓝牙服务	getBLEDeviceServices	deviceId: 连接的设备 ID	services: 设备服务列表
获取蓝牙特征值	getBLEDeviceCharacteristics	deviceId: 连接的设备 ID; serviceId: 蓝牙服务 uuid	characteristics: 目的服务中的特征值列表
数据上传	upload_request	url: 数据上传的 url files: 需要上传的文件列表	res: 调用结果
计算生物特征向量	embfordb	input: 原始特征 model: 加载的模型	embs: 特征向量 flag: 是否处理成功, 若失败到 embs 返回空值
判断两个生物特征是否是一个人	recognface	emb1: 第一个特征向量 emb2: 第二个特征向量	dis: 两个特征向量的欧式距离

5.7 错误/异常处理设计

5.7.1 错误/异常输出信息

- 1) 快应用被非法拉起时弹出提示信息。
- 2) 使用人脸检测的时候, 如果上传的图片没有检测到人脸, 则返回给快应用端未检测到人脸的提示信息。
- 3) 使用快应用上传数据时, 若设备 ID 未在数据库中找到, 则返回提示用户先注册设备信息的提示信息。

5.7.2 错误/异常处理对策

- 1) 快应用被拉起时,通过服务器参数判断是否为非法拉起,若为非法拉起则程序自动终止。
- 2) 未检测到人脸时,会返回布尔类型的 flag 值为 false,同时服务器会返回给快应用提示信息,快应用收到信息后,会进提示信息输出,并且再次拍摄,并进行图片上传。
- 3) 快应用上传数据是,若设备 ID 未在数据库中找到,则放弃本次操作,并返回给快应用端设备 ID 未注册的错误信息。快应用提示用户注册设备信息,并返回登录入口。
- 4) 服务器端的其他异常由框架处理并抛出并写入 log 文件,设计快应用的部分以 HttpResponse 的形式返回给快应用

5.8 系统配置策略

在项目的开发阶段因资金限制,认证授权服务器和 AI 服务器均部署在阿里云轻量应用服务器中,模型也部署进 Django 框架并部署 Nginx 以应对并发场景。

5.9 系统部署方案

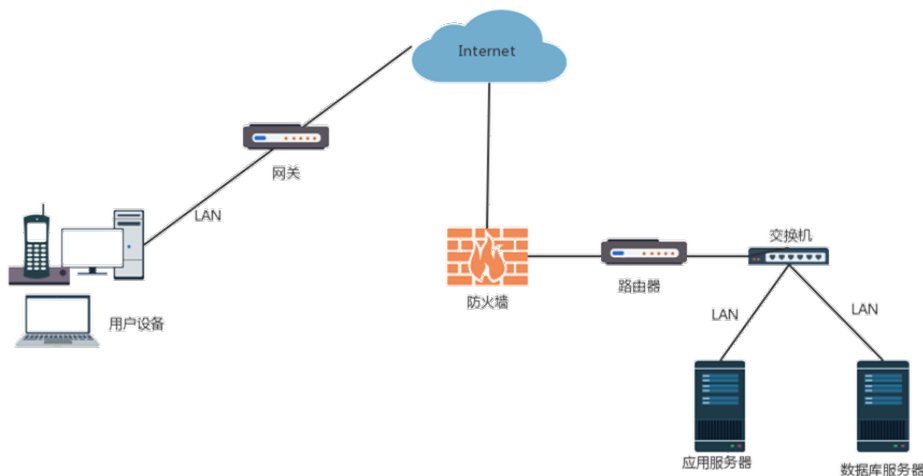


图 5.9.1 网络拓扑图

5.10 其他相关技术与方案

1) OAuth2.0 协议

OAuth 协议为用户资源的授权提供了一个安全的、开放而又简易的标准。与以往的授权方式不同之处是 OAuth 的授权不会使第三方触及到用户的帐号信息(如用户名与密码),即第三方无需使用用户用户名与密码就可以申请获得该用户资源的授权,因此 OAuth 是安全的。oAuth 是 Open Authorization 的简写。

OAuth2.0 是 OAuth 协议的延续版本,但不向前兼容 OAuth 1.0。OAuth 2.0 关注客户端开发者的简易性。要么通过组织在资源拥有者和 HTTP 服务商之间的被批准的交互

动作代表用户，要么允许第三方应用代表用户获得访问的权限。同时为 Web 应用，桌面应用和手机，和起居室设备提供专门的认证流程。

2) BLE 蓝牙通信技术

蓝牙低功耗（Bluetooth Low Energy，或称 Bluetooth LE、BLE，旧商标 Bluetooth Smart）也称低功耗蓝牙，是蓝牙技术联盟设计和销售的一种个人局域网技术，旨在用于医疗保健、运动健身、信标、安防、家庭娱乐等领域的新兴应用。相较经典蓝牙，低功耗蓝牙旨在保持同等通信范围的同时显著降低功耗和成本。

3) 人脸识别技术：

使用训练好的模型进行特征提取，将人脸信息转换成人脸特征向量存储在数据库中，在进行验证是，使用欧式距离计算人脸特征向量间的距离，并且以阈值来区分是否两个特征向量属于同一个人

6 数据库设计

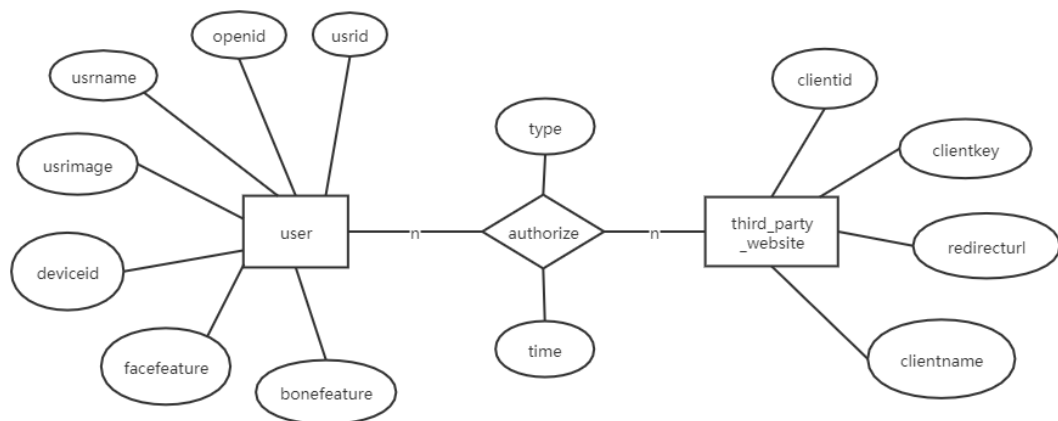


图 6.1 数据库 ER 图

usr

字段	类型	允许为空	默认
usrid (主键)	int	否	
openid	varchar(100)	否	
username	varchar(32)	是	NULL
usrimage	mediumblob	是	NULL

third_party_website

字段	类型	允许为空	默认
clientid (主键)	varchar(100)	否	
clientkey	varchar(100)	否	

redirecturi	varchar(100)	否	
clientname	varchar(100)	是	NULL

device

字段	类型	允许为空	默认
deviceid (主键)	varchar(20)	否	
usrid	int	否	

face

字段	类型	允许为空	默认
count (主键)	int	否	
usrid	int	否	
facefeature	text	否	

bone

字段	类型	允许为空	默认
count (主键)	int	否	
usrid	int	否	
bonefeature	text	否	

authorize

字段	类型	允许为空	默认
count (主键)	int	否	
type	enum('web','device','person')	否	'web'
clientid	varchar(100)	否	
usrid	int	否	
time	datetime	是	CURRENT_TIMESTAMP

7 详细设计

7.1 身份认证功能模块

7.1.1 功能描述

身份认证功能模块主要包含两个功能：面部认证和骨纹认证。

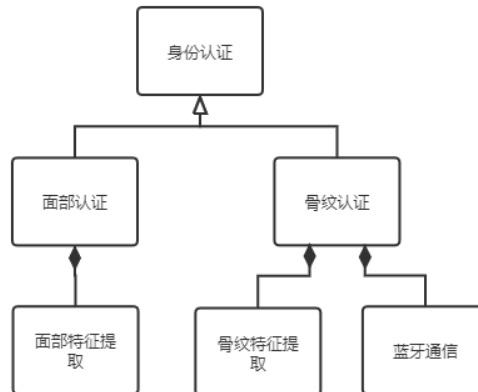


图 7.1.1.1 功能模块 UML 图

7.1.2 性能描述

用户自拍的图片分辨率与用户的具体机型有关。

蓝牙数据的传输时间不超过 1 秒

图片上传和接收认证结果的时间不超过 3 秒。

生物特征的识别准确度不低于 99%。

7.1.3 输入

用户的面部图片数据或骨纹数据。

7.1.4 输出

根据用户生物特征识别的结果不同，输出可能为多种情况：

- 1) 若用户上传的设备 ID 和生物特征在服务器中均正确匹配，则返回验证正确结果，即授权对象的登陆成功界面。
- 2) 若用户上传的设备 ID 与生物特征不匹配，或两者在服务器中未被找到，则返回验证失败结果，即取消本次登录并回到授权对象的登录入口。

7.1.5 程序逻辑

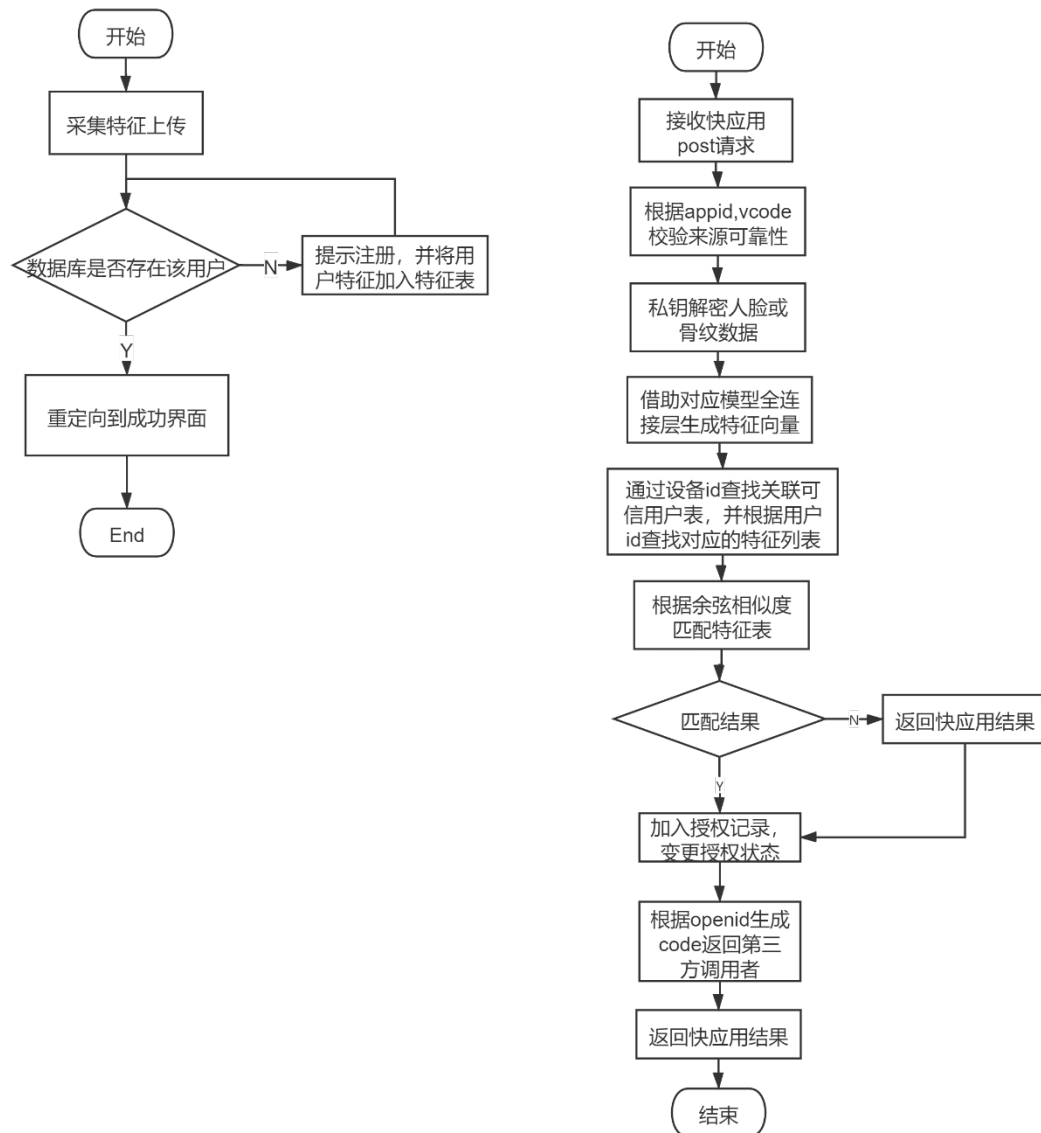


图 7.1.5.1 认证功能模块流程图

7.1.6 限制条件

用户允许快应用获取照相机权限；
用户允许快应用获取蓝牙权限；
蓝牙设备正常工作，并遵循我们规定的传输协议；
用户自拍图片需包括标准正脸全貌。

7.2 授权管理功能模块

7.2.1 功能描述

用户可以对以授权的对象进行管理,在通过身份验证的条件下可以取消对象的授权或更改授权信息。

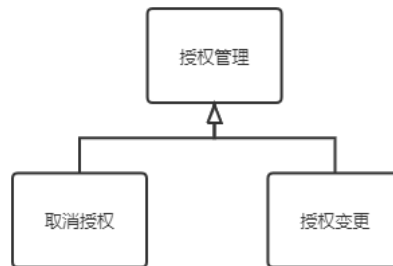


图 7.2.1.1 功能模块 UML 图

7.2.2 性能描述

用户取消已授权对象的授权并得到反馈结果的时间不超过 3 秒。

7.2.3 输入

用户在快应用上进行授权管理操作。

7.2.4 输出

根据用户进行授权管理时输入的操作不同,输出可能为多种情况:

- 2) 若用户取消某个对象的授权,则系统将该对象从授权列表中删除,并返回删除后的授权列表。
- 1) 若用户更改某个对象的授权信息,则系统将该对象的相应信息在数据库中更改,并返回更改后的授权列表。

7.2.5 程序逻辑

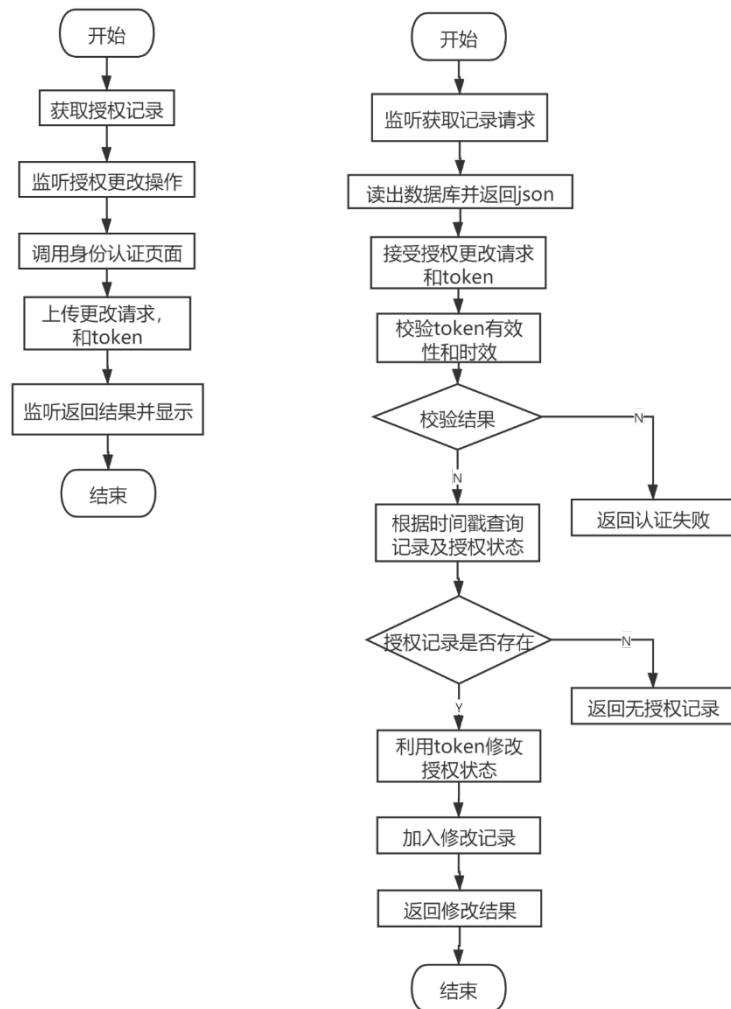


图 7.2.5.1 管理功能模块流程图

7.2.6 限制条件

用户在进行授权管理之前应先通过身份认证。