**简单概要设计文档**

目录

[1.应用程序架构图 2](#_Toc35031695)

[2.开发技术说明 3](#_Toc35031696)

[2.1 UI开发技术 3](#_Toc35031697)

[2.1.1 html 3](#_Toc35031698)

[2.1.2 CSS 3](#_Toc35031699)

[2.1.3 JS 4](#_Toc35031700)

[2.2 移动端开发技术 4](#_Toc35031701)

[2.2.1 Ionic 4](#_Toc35031702)

[2.3 桌面开发技术 4](#_Toc35031703)

[2.3.1 Electron 4](#_Toc35031704)

[2.4 Web端开发技术 4](#_Toc35031705)

[2.4.1 Vuetify 4](#_Toc35031706)

[2.5 服务器端开发技术 5](#_Toc35031707)

[2.5.1 Node.js 5](#_Toc35031708)

[2.6 显示层 5](#_Toc35031709)

[2.6.1 Ajax交互技术 5](#_Toc35031710)

[2.7 应用层 5](#_Toc35031711)

[2.7.1框架开发 5](#_Toc35031712)

[2.7.2 网络与通信 6](#_Toc35031713)

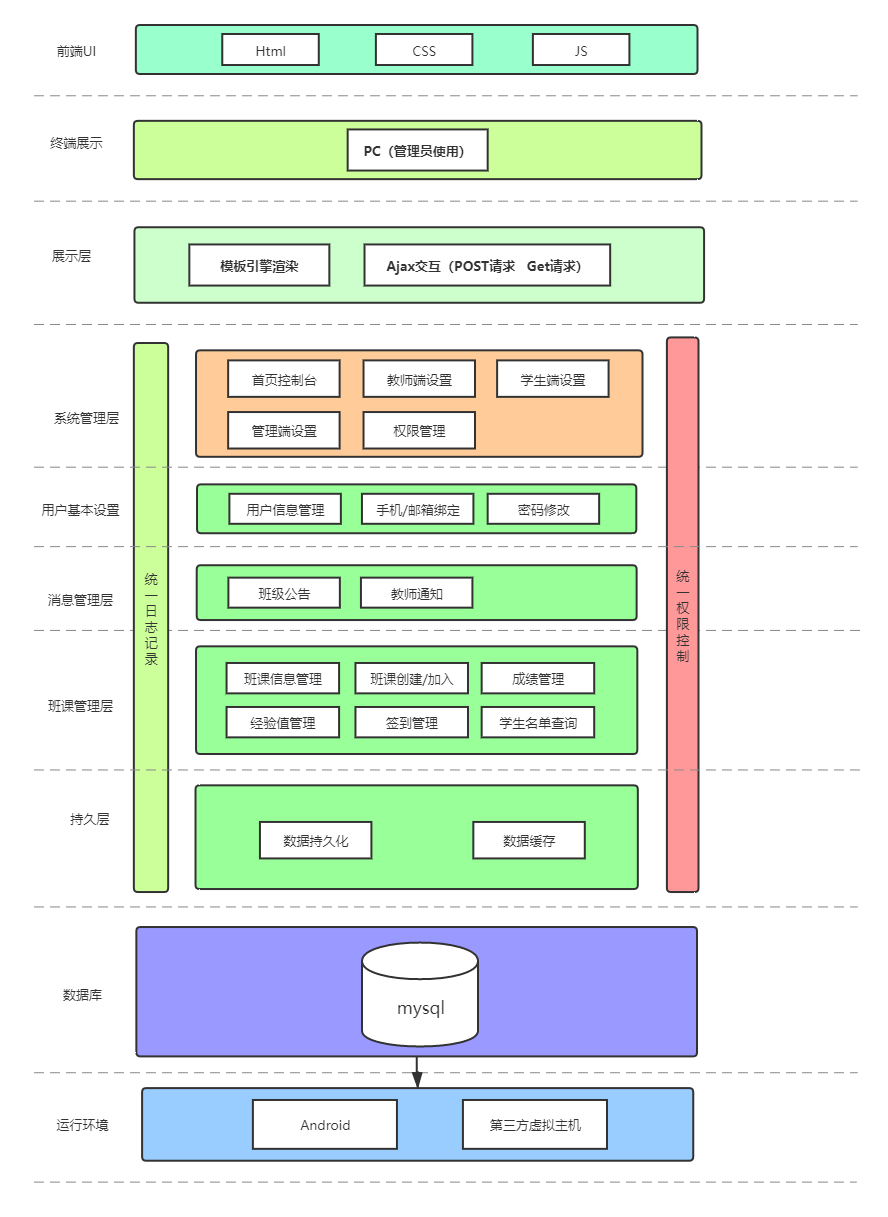
[2.7.3 缓存技术 6](#_Toc35031714)

[2.8 数据库 6](#_Toc35031715)

[2.8.1 SQL 6](#_Toc35031716)

[2.8.2 GreenDao框架 7](#_Toc35031717)

## 1.应用程序架构图

****

## 2.开发技术说明

### 2.1 UI开发技术

#### 2.1.1 html

html称为超文本标记语言，是一种标识性的语言。它包括一系列标签．通过这些标签可以将网络上的文档格式统一，使分散的Internet资源连接为一个逻辑整体。HTML文本是由HTML命令组成的描述性文本，HTML命令可以说明文字，图形、动画、声音、表格、链接等。超文本是一种组织信息的方式，它通过超级链接方法将文本中的文字、图表与其他信息媒体相关联。这些相互关联的信息媒体可能在同一文本中，也可能是其他文件，或是地理位置相距遥远的某台计算机上的文件。这种组织信息方式将分布在不同位置的信息资源用随机方式进行连接，为人们查找，检索信息提供方便。

#### 2.1.2 CSS

CSS的英文全称是Cascading Style Sheets(层叠样式表)，是一种用来表现HTML（标准通用标记语言的一个应用）或XML（标准通用标记语言的一个子集）等文件样式的计算机语言。CSS不仅可以静态地修饰网页，还可以配合各种脚本语言动态地对网页各元素进行格式化。CSS 能够对网页中元素位置的排版进行像素级精确控制，支持几乎所有的字体字号样式，拥有对网页对象和模型样式编辑的能力。

CSS具有以下特点：a) 丰富的样式定义。CSS提供了丰富的文档样式外观，以及设置文本和背景属性的能力；允许为任何元素创建边框，以及元素边框与其他元素间的距离，以及元素边框与元素内容间的距离；允许随意改变文本的大小写方式、修饰方式以及其他页面效果。b) 易于使用和修改。CSS可以将样式定义在HTML元素的style属性中，也可以将其定义在HTML文档的header部分，也可以将样式声明在一个专门的CSS文件中，以供HTML页面引用。总之，CSS样式表可以将所有的样式声明统一存放，进行统一管理。另外，可以将相同样式的元素进行归类，使用同一个样式进行定义，也可以将某个样式应用到所有同名的HTML标签中，也可以将一个CSS样式指定到某个页面元素中。如果要修改样式，我们只需要在样式列表中找到相应的样式声明进行修改。c) 多页面应用。CSS样式表可以单独存放在一个CSS文件中，这样我们就可以在多个页面中使用同一个CSS样式表。CSS样式表理论上不属于任何页面文件，在任何页面文件中都可以将其引用。这样就可以实现多个页面风格的统一。d) 层叠。简单的说，层叠就是对一个元素多次设置同一个样式，这将使用最后一次设置的属性值。例如对一个站点中的多个页面使用了同一套CSS样式表，而某些页面中的某些元素想使用其他样式，就可以针对这些样式单独定义一个样式表应用到页面中。这些后来定义的样式将对前面的样式设置进行重写，在浏览器中看到的将是最后面设置的样式效果。

#### 2.1.3 JS

JS是JavaScript的简称，是一种具有函数优先的轻量级，解释型或即时编译型的编程语言。虽然它是作为开发Web页面的脚本语言而出名的，但是它也被用到了很多非浏览器环境中，JavaScript 基于原型编程、多范式的动态脚本语言，并且支持面向对象、命令式和声明式（如函数式编程）风格。

JavaScript的组成部分有：1) ECMAScript，描述了该语言的语法和基本对象。2) 文档对象模型（DOM），描述处理网页内容的方法和接口。3) 浏览器对象模型（BOM），描述与浏览器进行交互的方法和接口。其基本特点有：1) 是一种解释性脚本语言（代码不进行预编译）。2) 主要用来向HTML（标准通用标记语言下的一个应用）页面添加交互行为。3) 可以直接嵌入HTML页面，但写成单独的js文件有利于结构和行为的分离。4) 跨平台特性，在绝大多数浏览器的支持下，可以在多种平台下运行（如Windows、Linux、Mac、Android、iOS等）。5) Javascript脚本语言同其他语言一样，有它自身的基本数据类型，表达式和算术运算符及程序的基本程序框架。Javascript提供了四种基本的数据类型和两种特殊数据类型用来处理数据和文字。而变量提供存放信息的地方，表达式则可以完成较复杂的信息处理。

### 2.2 移动端开发技术

#### 2.2.1 Ionic

Ionic是一个用来开发混合手机应用的，开源的，免费的代码库。可以优化html、css和js的性能，构建高效的应用程序，而且还可以用于构建Sass和AngularJS的优化。ionic会是一个可以信赖的框架。

### 2.3 桌面开发技术

#### 2.3.1 Electron

Electron是由Github开发，用HTML，CSS和JavaScript来构建跨平台桌面应用程序的一个开源库。 Electron通过将Chromium和Node.js合并到同一个运行时环境中，并将其打包为Mac，Windows和Linux系统下的应用来实现这一目的。

### 2.4 Web端开发技术

#### 2.4.1 Vuetify

Vuetify是一套用于构建用户界面的渐进式JavaScript框架,简写为Vue。与其它大型框架不同的是，Vue 被设计为可以自底向上逐层应用。Vue 的核心库只关注视图层，方便与第三方库或既有项目整合。其特点是：1) 易用。已经会了HTML,CSS,JavaScript？即刻阅读指南开始构建应用。2) 灵活。简单小巧的核心，渐进式技术栈，足以应付任何规模的应用。3)性能。20kb min+gzip 运行大小、超快虚拟 DOM 、最省心的优化。

### 2.5 服务器端开发技术

#### 2.5.1 Node.js

Node.js 就是运行在服务端的 JavaScript。

Node.js 是一个基于Chrome JavaScript 运行时建立的一个平台。

Node.js是一个事件驱动I/O服务端JavaScript环境，基于Google的V8引擎，V8引擎执行Javascript的速度非常快，性能非常好。

### 2.6 显示层

#### 2.6.1 Ajax交互技术

Ajax 即“Asynchronous Javascript And XML”（异步 JavaScript 和 XML），是指一种创建交互式、快速动态网页应用的网页开发技术，无需重新加载整个网页的情况下，能够更新部分网页的技术。通过在后台与服务器进行少量数据交换，Ajax 可以使网页实现异步更新。这意味着可以在不重新加载整个网页的情况下，对网页的某部分进行更新。Ajax过程:1) 创建XMLHttpRequest对象,也就是创建一个异步调用对象。2) 创建一个新的HTTP请求,并指定其请求的方法、URL及验证信息。3) 设置响应 HTTP 请求状态变化的函数。4) 发送 HTTP 请求。5) 获取异步调用返回的数据。6) 使用 JavaScript 和 DOM 实现局部刷新。

### 2.7 应用层

#### 2.7.1框架开发

EventBus是一个消息总线，以观察者模式实现，用于简化程序的组件、线程通信，可以轻易切换线程、开辟线程。在做项目的时候往往需要应用程序内各组件间、组件与后台线程间的通信。比如耗时操作，等耗时操作完成后通过Handler或Broadcast将结果通知给UI，N个Activity之间需要通过Listener通信。其实这些都可以通过EventBus轻松实现，EventBus通过发布/订阅(publish/subscribe)方式来管理事件总线。EventBus通过注解和反射机制 将订阅者连同订阅函数保存起来，然后在发送订阅的时候 遍历订阅函数数组进行调用，其实从这方面就可以EventBus执行效率多少会受到一点影响。

#### 2.7.2 网络与通信

ContentProvider来获取和更改手机系统中通讯录、短信的数据。当应用继承ContentProvider类，并重写该类用于提供数据和存储数据的方法，就可以向其他应用共享其数据。虽然使用其他方法也可以对外共享数据，但数据访问方式会因数据存储的方式而不同，如：采用文件方式对外共享数据，需要进行文件操作读写数据；采用sharedpreferences共享数据，需要使用sharedpreferences API读写数据。而使用ContentProvider共享数据的好处是统一了数据访问方式。

基于Xmpp通信协议的IM（即时通信）开发。xmpp可以理解为可扩展的消息和出席协议（eXtensible Messageing and Presence Protocol）.出席即可以理解为用户的在线的状态，消息则是服务器与客户端互相通信的消息；常见的xmpp服务器有openfire、Ejabberd等，这里我们用的是openfire。xmpp在网络上通信的每个实体都有统一的ID表示，即为JID(jabber identifier);JID有很多不同的形式，通常类似于邮件的形式。在xmpp中，数据的传输和识别是通过在一个xmpp流上发送和接收xmpp节点完成的，这三个节点分别是presence、message、iq;这三个节点都有通用的属性from、to、type、id；xmpp传输数据也是通过xml文档实现的。

#### 2.7.3 缓存技术

[ASimpleCache](http://www.codeceo.com/article/asimplecache-android-cache.html)是一款基于Android的轻量级缓存框架，它只有一个Java文件，ASimpleCache基本可以缓存常用的Android对象，包括但不限于普通字符串，JSON对，经过序列化的Java对象，字节数组。

ASimpleCache的特点：轻量级，只有一个Java文件；完整而灵活的配置，可以配置缓存路径，缓存大小，缓存数量，缓存超时时间等；超时缓存自动失效，并从内存中自动删除；多进程的支持。在Android开发中，我们可以用ASimpleCache来替换SharePreference配置文件，特别是如果你的应用经常要从互联网上读取数据，那么利用ASimpleCache可以缓存这些请求数据，等一段时间失效后再去重新读取，这样可以减少客户端流量，同时减少服务器并发量。

### 2.8 数据库

#### 2.8.1 SQL

**SQL语言**，是结构化查询语言(Structured Query Language)的简称。SQL语言是一种数据库查询和[程序设计](https://baike.so.com/doc/3745498.html)语言，用于存取数据以及查询、更新和管理关系[数据库系统](https://baike.so.com/doc/3054063.html)；同时也是数据库[脚本文件](https://baike.so.com/doc/547484.html)的扩展名。

SQL语言是高级的非过程化编程语言，允许用户在高层数据结构上工作。它不要求用户指定对数据的存放方法，也不需要用户了解具体的数据存放方式，所以具有完全不同底层结构的不同数据库系统可以使用相同的结构化查询语言作为数据输入与管理的接口。SQL语言语句可以嵌套，这使他具有极大的灵活性和强大的功能。

#### 2.8.2 GreenDao框架

在Android开发过程中，有时候我们需要使用SQLite数据库去本地存储一些临时文件，之前，我们的做法是通过SQLiteOpenHelper实现创建数据库，以及迭代开发中的数据库数据内容字段变更时处理。通常我们在使用GreenDao的时候，我们只需定义数据模型，GreenDao框架将创建数据对象(实体)和DAO(数据访问对象)，能够节省部分代码。不向性能妥协，使用了GreenDao，大多数实体可以以每秒几千个实体的速率进行插入，更新和加载。GreenDao支持加密数据库来保护敏感数据，具有微小的依赖库，如果需要，实体是可以被激活的。而活动实体可以透明的解析关系（我们要做的只是调用getter即可），并且有更新、删除和刷新方法，以便访问持久性功能。允许将协议缓冲区(protobuf)对象直接保存到数据库中。如果您通过protobuf通话到服务器，则不需要另一个映射。常规实体的所有持久性操作都可以用于protobuf对象。所以，相信这是GreenDao的独特之处。