目录

[1引言 2](#_Toc66124031)

[1.1目的 2](#_Toc66124032)

[1.2 背景 2](#_Toc66124033)

[1.3定义 2](#_Toc66124034)

[2 项目概述 4](#_Toc66124035)

[2.1 产品结构图 4](#_Toc66124036)

[2.2 产品信息结构图 4](#_Toc66124037)

[2.3 用户 5](#_Toc66124038)

[2.4 业务流程 6](#_Toc66124039)

[2.5 全局说明 7](#_Toc66124040)

[3功能详细需求 7](#_Toc66124041)

[3 功能详细需求 7](#_Toc66124042)

[3.1 登录 7](#_Toc66124043)

[3.1.1 验证码登录 7](#_Toc66124044)

[3.1.2 密码登录 8](#_Toc66124045)

[3.2 注册 8](#_Toc66124046)

[3.2.1 注册 8](#_Toc66124047)

[3.3 班课 9](#_Toc66124048)

[3.3.1 我创建的 9](#_Toc66124049)

[3.3.2 我加入的 9](#_Toc66124050)

[3.3.3 创建班课 10](#_Toc66124051)

[3.3.4 查看班课成员 10](#_Toc66124052)

[3.4 我的 11](#_Toc66124053)

[3.4.1 用户信息 11](#_Toc66124054)

[4非功能需求 11](#_Toc66124055)

[4.1安全需求 11](#_Toc66124056)

[4.2 性能需求 11](#_Toc66124057)

[4.2.1动态数值需求 11](#_Toc66124058)

[4.2.2 静态数值需求 12](#_Toc66124059)

[4.3 兼容性需求 12](#_Toc66124060)

[4.4可用性需求 12](#_Toc66124061)

1引言

## 1.1目的

此概要设计说明书是为了说明整个项目的体系架构，以及确定开发该项目的技术选型。并对数据库进行初步的设计。

## 1.2 背景

我们现在理解的传统教学方式主要是指利用板书和多媒体进行课堂教学，学生提交作业主要是通过纸质资料和电子资料进行，教学互动环节也主要是面对面的互动为主，局限性较大。传统教学方式下作业提交及教学互动方式如下:

在传统教学方式下，学生提交作业主要是通过纸质资料和电子资料进行，教师批改作业也只能是将纸质作业带在身边，或者携带电脑才能进行电子作业的批改。这在无形中给教师带来了不便，很多时候由于作业不便于携带只能在办公室修改;同时，现在的学生，尤其是高职院校的学生自制力很差，上课经常不带书不带笔，课后更不会去完成老师布置的作业。所以，有时传统模式下的作业提交变得非常困难，效果不是很好。

利用板书与多媒体进行教学时，课堂点名、课堂提问、小组讨论等教学互动环节基本上也都是在课堂上进行，而且教学的资源基本上只有教材和教师的多媒体课件,学生的思维容易受到书本知识的约束，课堂互动效果不是很好，学生的积极性也较差。而现在通过移动终端等智能设备和互联网的应用，就可以实现全方位、立体式的交互，实现师生交互、生生交互！真正实现课前、课中、课后、课内、课外、线上、线下的全场景立体交互。同时，交互的内容也会发生一些改变，除了传统的一些内容外增加了微课、富媒体，同时也跨越了时间和地点。

最后，是评测方式的改变。评测方式大家都知道以前去评价一个学生的好坏，就是以分数为唯一的评价标准，我们说这种评级是一种以偏概全的，而且评价的信息比较滞后，同时仅仅是在认知层面的评价，是一种结果性的评价。如今，互联网通过大数据分析及应用等技术手段，记录教育教学过程，把你的行为数据采集下来，实现全过程的动态评价。这种评价就是从结果性评价转为过程性评价，体现我们的综合素质评价。

## 1.3定义

API:应用程序接口（Application Programming Interface），是一些预先定义的函数，或指软件系统不同组成部分衔接的约定。

app:应用程序，Application的缩写，一般指手机软件。

MTBF:即平均故障间隔时间，英文全称是“Mean Time Between Failure”。是衡量一个产品（尤其是电器产品）的可靠性指标。

Web:即全球广域网（World Wide Web），也称为万维网，它是一种基于超文本和HTTP的、全球性的、动态交互的、跨平台的分布式图形信息系统。

Tomcat：是Java领域最著名的开源web容器，简单，易用，稳定性极好，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试JSP程序的首选。Tomcat不仅提供了web容器的基本功能，还支持JAAS和JNDI绑定等，而且其完全是纯Java实现，与平台无关。

MySQL：是一个开放源码的小型关联式数据库管理系统，开发者为瑞典MySQLAB公司。目前MySQL 被广泛地应用在Internet上的中小型网站中。由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，许多中小型网站为了降低网站总体拥有成本而选择了MySQL作为网站数据库。

Android: 安卓是一种基于[Linux](https://baike.baidu.com/item/Linux)内核（不包含[GNU](https://baike.baidu.com/item/GNU/671972)组件）的自由及开放源代码的[操作系统](https://baike.baidu.com/item/%E6%93%8D%E4%BD%9C%E7%B3%BB%E7%BB%9F/192)。主要使用于[移动设备](https://baike.baidu.com/item/%E7%A7%BB%E5%8A%A8%E8%AE%BE%E5%A4%87/9157757)，如[智能手机](https://baike.baidu.com/item/%E6%99%BA%E8%83%BD%E6%89%8B%E6%9C%BA/94396)和[平板电脑](https://baike.baidu.com/item/%E5%B9%B3%E6%9D%BF%E7%94%B5%E8%84%91/1348389)，由美国[Google](https://baike.baidu.com/item/Google)公司和[开放手机联盟](https://baike.baidu.com/item/%E5%BC%80%E6%94%BE%E6%89%8B%E6%9C%BA%E8%81%94%E7%9B%9F/9064338)领导及开发。Android操作系统最初由[Andy Rubin](https://baike.baidu.com/item/Andy%20Rubin)开发，主要支持[手机](https://baike.baidu.com/item/%E6%89%8B%E6%9C%BA/6342)。2005年8月由Google收购注资。2007年11月，Google与84家硬件制造商、软件开发商及电信营运商组建开放手机联盟共同研发改良Android系统。随后Google以Apache开源许可证的授权方式，发布了Android的源代码。第一部Android智能手机发布于2008年10月。Android逐渐扩展到平板电脑及其他领域上，如[电视](https://baike.baidu.com/item/%E7%94%B5%E8%A7%86/228945)、[数码相机](https://baike.baidu.com/item/%E6%95%B0%E7%A0%81%E7%9B%B8%E6%9C%BA/277472)、[游戏机](https://baike.baidu.com/item/%E6%B8%B8%E6%88%8F%E6%9C%BA/315328)、[智能手表](https://baike.baidu.com/item/%E6%99%BA%E8%83%BD%E6%89%8B%E8%A1%A8/71070)等。2011年第一季度，Android在全球的市场份额首次超过[塞班系统](https://baike.baidu.com/item/%E5%A1%9E%E7%8F%AD%E7%B3%BB%E7%BB%9F/8506777)，跃居全球第一。

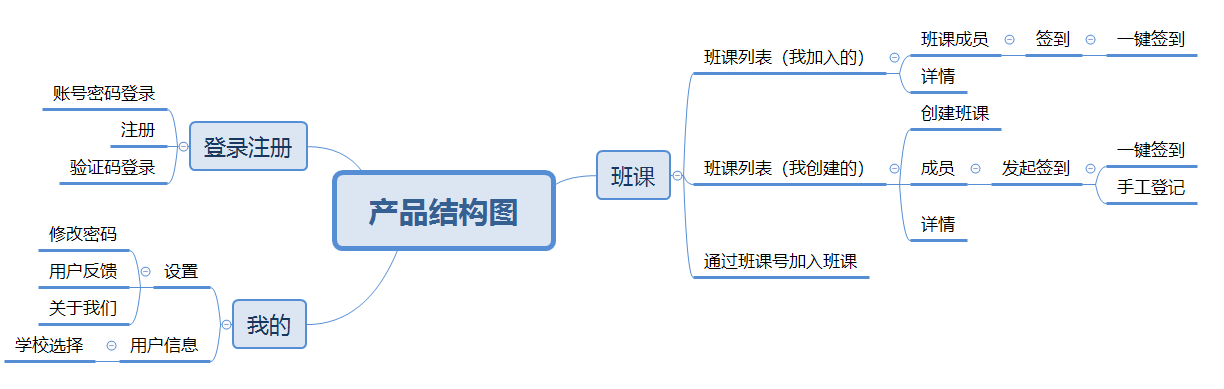
Android Studio：Android Studio 是谷歌推出的一个Android集成开发工具，基于[IntelliJ IDEA](https://baike.baidu.com/item/IntelliJ%20IDEA" \t "_blank). 类似 [Eclipse](https://baike.baidu.com/item/Eclipse/61703) [ADT](https://baike.baidu.com/item/ADT)，Android Studio 提供了集成的 Android 开发工具用于开发和调试。

Vue.js: Vue.js是一套构建用户界面的渐进式框架。与其他重量级框架不同的是，Vue 采用自底向上增量开发的设计。Vue 的核心库只关注视图层，并且非常容易学习，非常容易与其它库或已有项目整合。另一方面，Vue 完全有能力驱动采用单文件组件和Vue生态系统支持的库开发的复杂单页应用。Vue.js 的目标是通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件 。Vue.js 自身不是一个全能框架——它只聚焦于视图层。因此它非常容易学习，非常容易与其它库或已有项目整合。另一方面，在与相关工具和支持库一起使用时 ，Vue.js 也能完美地驱动复杂的单页应用。

Element:Element UI 是一套采用 Vue 2.0 作为基础框架实现的组件库,一套为开发者、设计师和产品经理准备的基于 Vue 2.0 的组件库,提供了配套设计资源,帮助网站快速成型。

# 2 项目概述

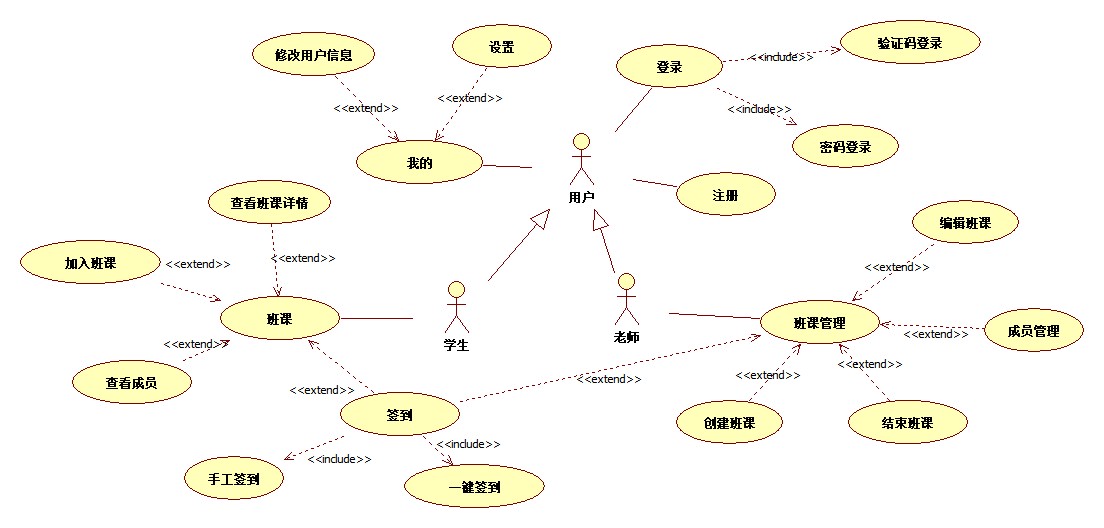
## 2.1 产品结构图



## 2.2 产品信息结构图

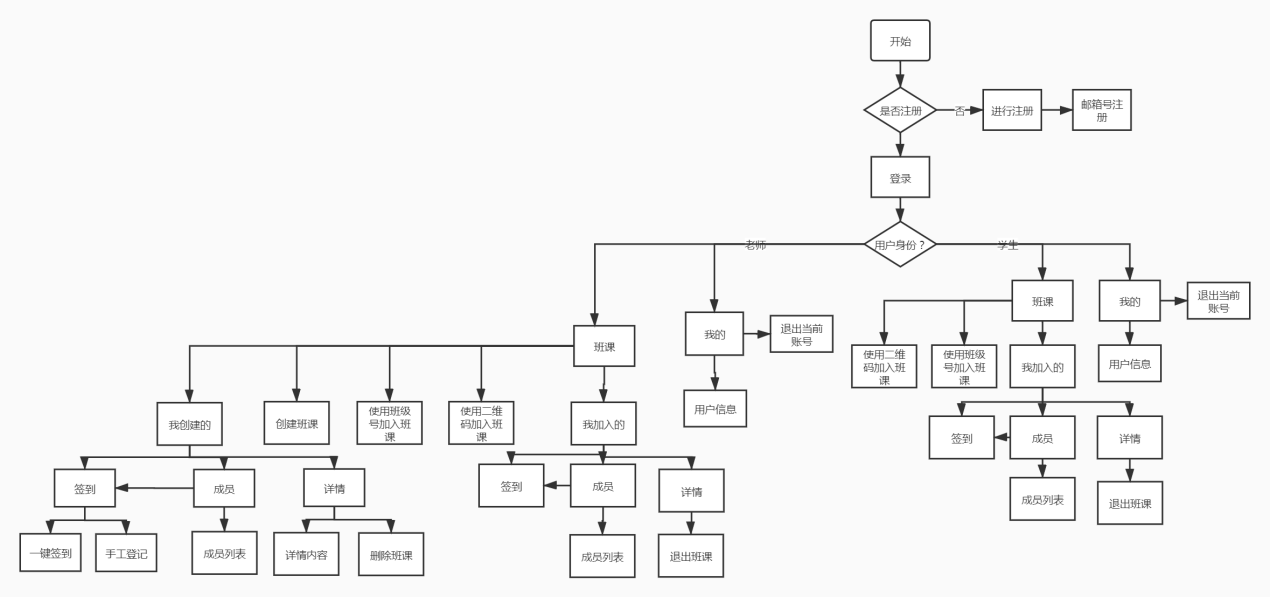


## 2.3 用户



|  |  |  |  |
| --- | --- | --- | --- |
| 用户角色 | 频道 | 功能 | 详细描述 |
|  | 我的 | 用户信息 | 用户可以修改自己的基本信息。 |
| 创建班课 | 创建班课 | 老师可以创建一个班课，设置班课的基本信息，系统为会为新建的班课创建一个班课号，学生可以根据这个班课号加入班级。 |
| 班课列表 | 查看自己创建的所有班课。 |
| 成员 | 老师选择班课后，可以看到本班课中的成员列表，可以查看成员的详细信息。 |
| 班课详情 | 班课详情包括班课基本信息和一些操作组成。 |
| 签到管理 | 每个班课都有一个签到入口，老师可以使用一键签到（距离签到）、手工签到两种签到方式发起签到。 |
| 学生 | 我的 | 数据信息 | 显示经验值等数据信息。 |
| 用户信息 | 用户可以修改自己的基本信息。 |
| 班课 | 加入班课 | 用户根据老师给的班课号或者二维码，加入到相应的班课。 |
| 班课列表 | 显示用户加入的所有班课，在列表中可以通过签到按钮进行签到。 |
| 成员 | 成员列表显示的是加入本班课的成员。 |
| 签到 | 老师发起签到后，用户可以参与签到。 |
| 班课详情 | 班课详情包括班课基本信息和一些操作组成。 |

## 2.4 业务流程



## 2.5 全局说明

数据加载的时候经常出现几种状态，为了给用户一些良好的体验，就必须有相对应的提示：

1. 页面加载/提交中：加载；
2. 加载/刷新/提交成功：成功提示；
3. 加载/刷新/提交失败：失败原因短提示，系统提示后淡出。

# 3 功能详细需求

## 3.1 登录

### 3.1.1 验证码登录

**用户场景：**程序第一次运行时，用户进入登录页面，用户填写有效的邮箱，之后获取验证码登录，登录成功后进入应用程序首页。用户登录成功后，就能够使用应用程序的所有功能。30天之内再使用软件都不需要再次登录，用户直接进入程序首页。如果之前用户没有登录过，或者登录时间已经过期，则运行该程序时，进入登录页面。

**优先级：**无

**业务流程：**无

**输入/前置条件：**用户已完成注册，用户未登录或者登录时间已过期。

**界面原型：**

**交互**：如果用户之前登录过，账户输入框中默认显示用户之前登录时用过的账号。

**字段：**

|  |  |  |
| --- | --- | --- |
| **字段** | **说明** | **数据来源** |
| 账号 | 必填，邮箱 | 用户输入 |
| 密码 | 必填 | 用户输入 |

**输出/后置条件：**登录成功后，保存用户的登录状态和当前的登录时间，页面跳转到应用程序首页。

**版本**：无**。**

### 3.1.2 密码登录

**用户场景：**程序第一次运行时，用户进入登录页面，用户填写邮箱和密码，登录成功后进入应用程序首页。用户登录成功后，就能够使用应用程序的所有功能。30天之内再使用软件都不需要再次登录，用户直接进入程序首页。如果之前用户没有登录过，或者登录时间已经过期，则运行该程序时，进入登录页面。

**优先级：**无

**业务流程：**无。

**输入/前置条件：**用户已完成注册，用户未登录或者登录时间已过期。

**交互：**如果用户之前登录过，账户输入框中默认显示用户之前登录时用过的账号。填写邮箱后，点击获取验证码按钮，用户收到一条验证码。

**字段：**

|  |  |  |
| --- | --- | --- |
| **字段** | **说明** | **数据来源** |
| 账号 | 必填，邮箱 | 用户输入 |
| 验证码 | 必填 | 用户输入 |

**输出/后置条件：**登录成功后，保存用户的登录状态和当前的登录时间，页面跳转到应用程序首页。

**版本：**无

## 3.2 注册

### 3.2.1 注册

**用户场景：**用户注册之后才能有权限使用软件的功能。用户在登录页点击“注册”按钮进入注册页。用户填写邮箱，然后点击“获取验证码”，稍候片刻，会有一条短信发送到你的邮箱上，输入验证码之后，点击“注册”按钮，然后为自己的账号设置一个密码。

**优先级：**无。

**业务流程：**用户先填写邮箱，邮箱验证有效之后，点击获取验证码，用户填入有效验证码，接下来用户为自己的账号设置一个密码，密码填写后会让用户重复输入密码，两次密码都验证成功后，用户注册成功，进到首页。

**输入/前置条件：**无。

**界面原型：**

**字段：**

|  |  |  |
| --- | --- | --- |
| **字段** | **说明** | **数据来源** |
| 账号 | 必填，邮箱，且邮箱验证是有效的且未注册过。 | 用户输入 |
| 短信验证码 | 必填，用户填写邮箱后获取 | 用户输入 |
| 登录密码 | 必填，长度为6-16位，必须至少包含英文、字符、数字中两种。 | 用户输入 |
| 确认密码 | 必填，必须与登录密码一致。 | 用户输入 |

**输出/后置条件：** 注册成功后，数据保存到数据库中，页面跳转到首页。

**版本：**无

## 3.3 班课

### 3.3.1 我创建的

**用户场景：**

用户登录后进入班课页面，选择“我创建的”模块，可以看到该用户所创建的班级，可以发起签到，可选择进行一键签到、手工登记两种模式。一键签到实际是距离签到，老师在WEB端设置距离后，在距离范围内的用户才可以签到成功。

点击班级详细界面，在成员一栏可以发起签到，签到内容与上述一致，并可以查看历史签到记录。接下来展示班课中成员列表，列表默认按经验值高低排序，可以切换为按学号排行以及相应成员搜索操作。

在详情一栏，可以查看班课的基本信息，可以进行编辑修改操作，以及删除班课。

**业务流程：**无

**界面原型：**

**输入/前置条件：**用户身份为教师。

**输出/后置条件：**无。

### 3.3.2 我加入的

**用户场景：**用户登录后进入班课页面，选择“我加入的”模块，可以看到该用户所加入的班级，对老师发起的签到可点击签到操作。点击班级详细界面，在成员一栏也可进行参与签到，接下来显示成员总数及每个成员列表，点击可以查看成员的详细内容。点击详情一栏可以查看所加入班课的详细信息，也可进行退出班课操作。

**业务流程：**无。

**输入/前置条件：**用户身份为学生。

**输出/后置条件：**无。

### 3.3.3 创建班课

**用户场景：**老师用户登录后进入班课页面，点击右上角加号，可进行创建班课的操作。只有当用户身份为老师时才有创建班课的权限，可以设置班级，课程，学期以及更多详细内容来进行班课的创建。

学生用户登录后进入班课页面，点击右上角加号，可使用班课号以及扫描二维码加入班课。只有当用户身份为学生时才有创建班课的权限。创建好的班课都会有相应的班课号，使用班课号来加入班课，输入班课号，然后点击下一步，可看到该课程名及简略信息，点击加入即完成班课申请操作；或使用扫描班课号的二维码来加入班课，扫描成功后可看到该课程名及简略信息，点击加入即完成班课申请操作。

**业务流程：**无。

**输入/前置条件：**用户身份为老师/学生。

**输出/后置条件：**无。

### 3.3.4 查看班课成员

**用户场景：**

教师用户登录后进入班课页面，可以看到该用户所创建的班级。选中其中一个班课，默认进入活动页面。点击底部导航栏进入成员界面，展示班课中成员列表，列表默认按经验值高低排序，可以切换为按学号排行以及相应成员搜索操作，每条成员记录包含以下信息：姓名、头像、性别、学号、排名、经验值。

**业务流程：**无。

**输入/前置条件：**用户身份为老师。

**输出/后置条件：**无。

**界面原型**

## 3.4 我的

### 3.4.1 用户信息

**用户场景：**用户登录后进入“我的”页面，再点击昵称一栏，进入用户信息页面。在用户信息页面可以编辑自己的基本信息，包括头像、姓名、昵称、出生年份、性别、你所在的学校及院系、你的身份、学号/工号等内容，编辑完毕后点击保存即可。

**业务流程：**无。

**输入/前置条件：**用户身份为学生/老师。

**界面原型：**

**输出/后置条件:**无。

# 4非功能需求

4.1安全需求

（1）所有用户属性信息的传输，都需要在接口层进行数据加密。

（2）安卓系统要混淆过代码，以防被竞品开发者或攻击者破解代码。

（3）移动端系统的用户应该注册后才能登录，防止恶意刷包。

* 1. 性能需求

4.2.1动态数值需求

|  |  |
| --- | --- |
| 功能 | 性能需求 |
| 首页 | 用户从登录页面输入账号密码点击登录到进入到首页的时间不超过1s。 |
| 单页面 | 单页面加载时间不超过2s，若2s后无响应，需要给出提示：网络繁忙，请稍后重试。且点击页面可以重新加载。 |
| 用户数据 | 个人信息中的姓名，性别，联系方式数据大小范围不超过16个字节 |
| API请求 | API请求的平均响应时间应低于1s, WEB首页打开速度5s以下，web登陆速度 15s以下。 |

4.2.2 静态数值需求

后台管理系统支持10个用户同时在线。

4.3 兼容性需求

该移动系统应能够兼容各种主流手机品牌，如小米、华为、vivo、一加手机等，各手机均可正常使用，文字、图片等信息显示正常，无显示异常。

4.4可用性需求

（1）完成同一功能或任务的元素放在集中位置，减少鼠标移动的距离。

（2）需要避免用户重复点击。

（3）同一界面上的控件数最好不要超过10个，多于10个时可以考虑使用分页界面显示。

（4）需要保证即时响应用户的操作。

（5）需要避免多个用户操作时产生冲突。