

到云

**软件需求说明书**

作 者： 黄旭、杨立坚、李鸣、林炳辉、付乐乐

完成日期：

签 收 人：

签收日期：

修改情况记录：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本号 | 修改批准人 | 修改人 | 安装日期 | 签收人 |
| V1.0 |  |  |  |  |
| V2.0 |  |  |  |  |
|  |  |  |  |  |

目录

[**1 引言** 3](#_Toc34228679)

[**1.1 目的** 3](#_Toc34228680)

[**1.2 背景** 3](#_Toc34228681)

[**1.3 定义** 3](#_Toc34228682)

[**2 产品概述** 4](#_Toc34228683)

[**2.1 产品描述** 4](#_Toc34228684)

[**2.2 产品结构图** 4](#_Toc34228685)

[**2.3 产品全局结构图** 5](#_Toc34228686)

[**2.4 用户** 5](#_Toc34228687)

[**2.5 业务流程** 6](#_Toc34228688)

[**3 业务功能** 8](#_Toc34228689)

[**3.1 注册与登录** 8](#_Toc34228690)

[**3.1.1注册与登录** 8](#_Toc34228691)

[**3.2 班课** 12](#_Toc34228692)

[**3.2.1 创建班课** 12](#_Toc34228693)

[**3.2.2 加入班级** 15](#_Toc34228694)

[**3.2.3 查看班级** 17](#_Toc34228695)

[**3.2.3 作业** 19](#_Toc34228696)

[**3.2.4 签到** 23](#_Toc34228697)

[**3.2.5 搜索** 27](#_Toc34228698)

[**3.3 发现** 28](#_Toc34228699)

[**3.3.1课程圈** 28](#_Toc34228700)

[**3.4 我的** 30](#_Toc34228701)

[**3.4.1 我的各功能** 30](#_Toc34228702)

[**4 非功能需求** 37](#_Toc34228703)

**1 引言**

**1.1 目的**

目前高校大部分都还是采用任课教师上课点名的形式进行考勤点名工作，且多数线上作业需要通过单独收集的形式，结合高校课堂学生数量较多的情况，这样的形式是耗时耗力，大大降低了课堂的效率。在信息化的今天，能有效的借助网络才能更好的提高办事效率。因此，“到云”这款软件能够有效的针对上述问题进行优化。

**1.2 背景**

软件/产品名称：到云

提出者：池老师

开发者：黄旭、杨立坚、李鸣、林炳辉

用户：各大高校的老师与学生

**1.3 定义**

终端：表示最终使用本产品的机器，例如手机。

**2 产品概述**

**2.1 产品描述**

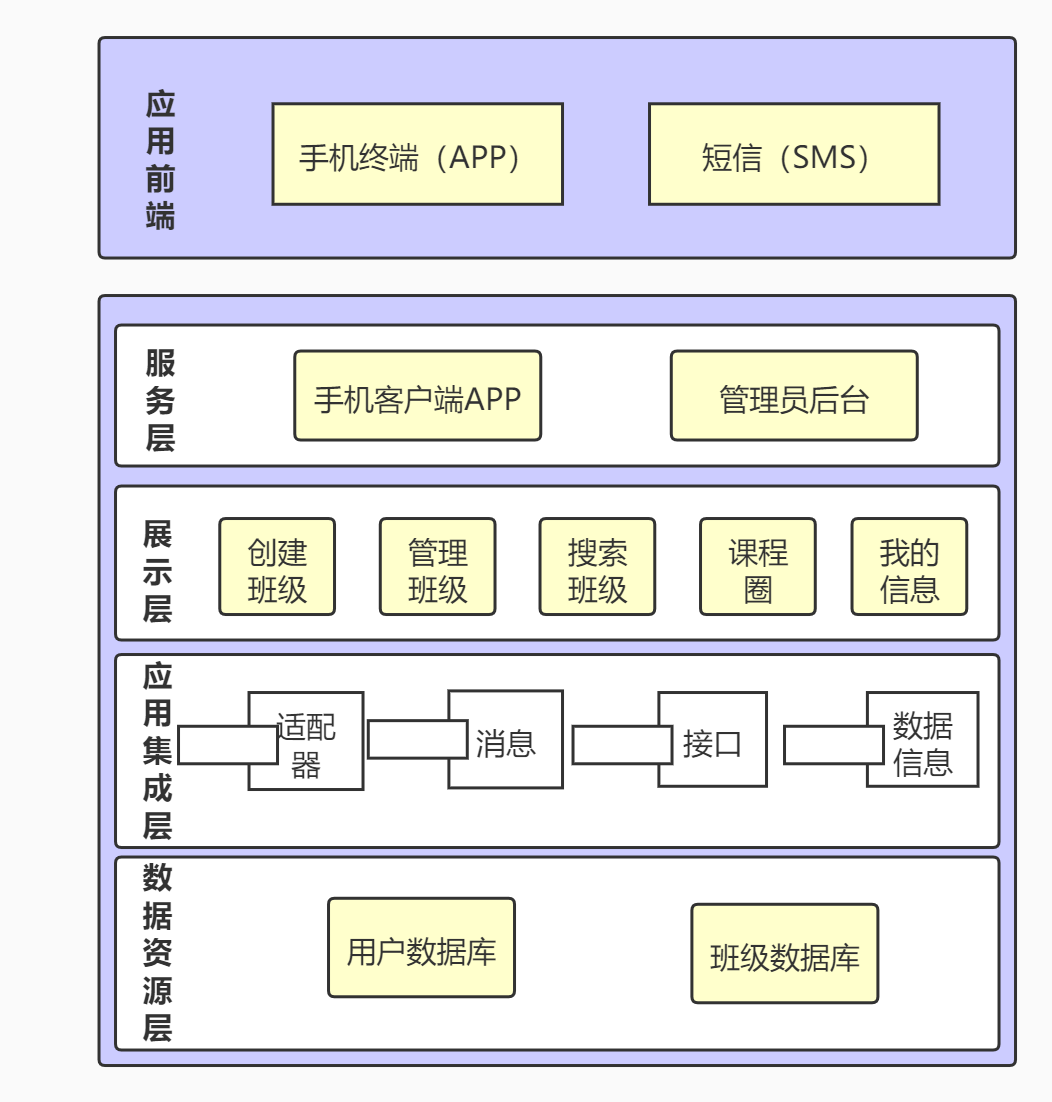
应用目标：针对高校教学中，人数多造成的签到耗费时间、线上作业没有合适的上交途径等问题，本产品致力于解决此类问题，提高高校课堂教学效率。

范围：该产品主要针对课堂签到、作业的发布与收集以及一切课堂圈内的讨论，不牵涉教学的具体内容。

开发背景：开发此产品主要是为了一个完成作业，一个是为了锻炼团队成员的文档写作能力和开发能力。

**2.2 产品结构图**

产品结构图如下图所示：



其中，包括以下几个结构：

应用前端：手机终端（APP）、短信（SMS）

服务层：手机客户端APP、管理员后台

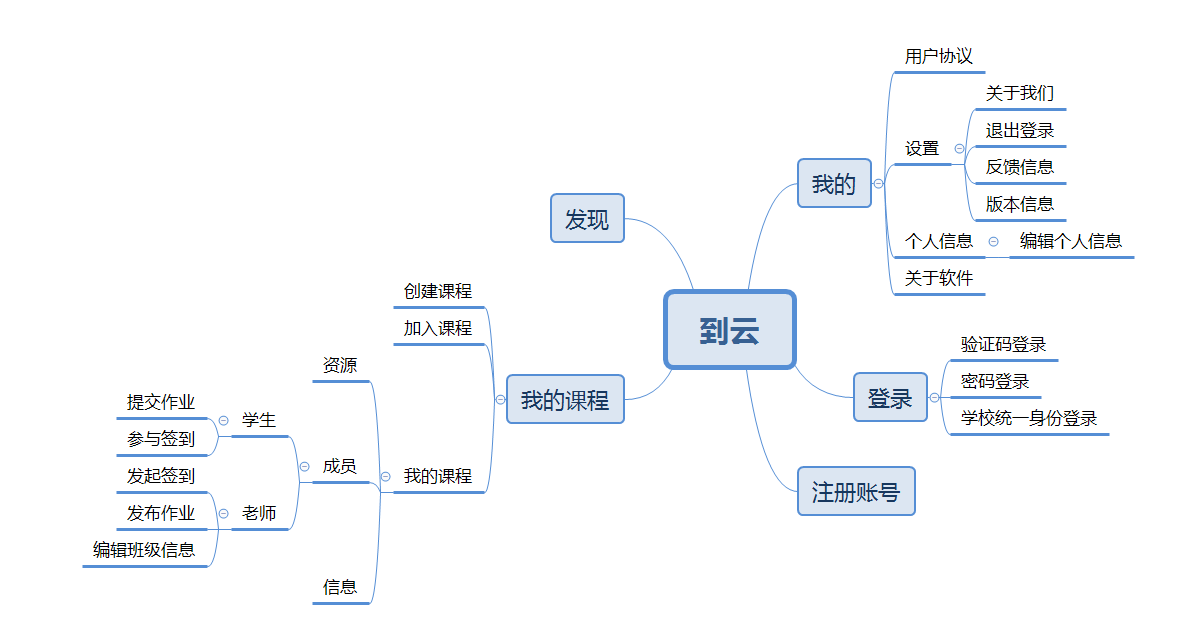
展示层：创建班级、管理班级、搜索班级、课程圈、我的信息

应用集成层：适配器、消息、接口、数据信息

数据资源层：用户数据库、班级数据库

**2.3 产品全局结构图**

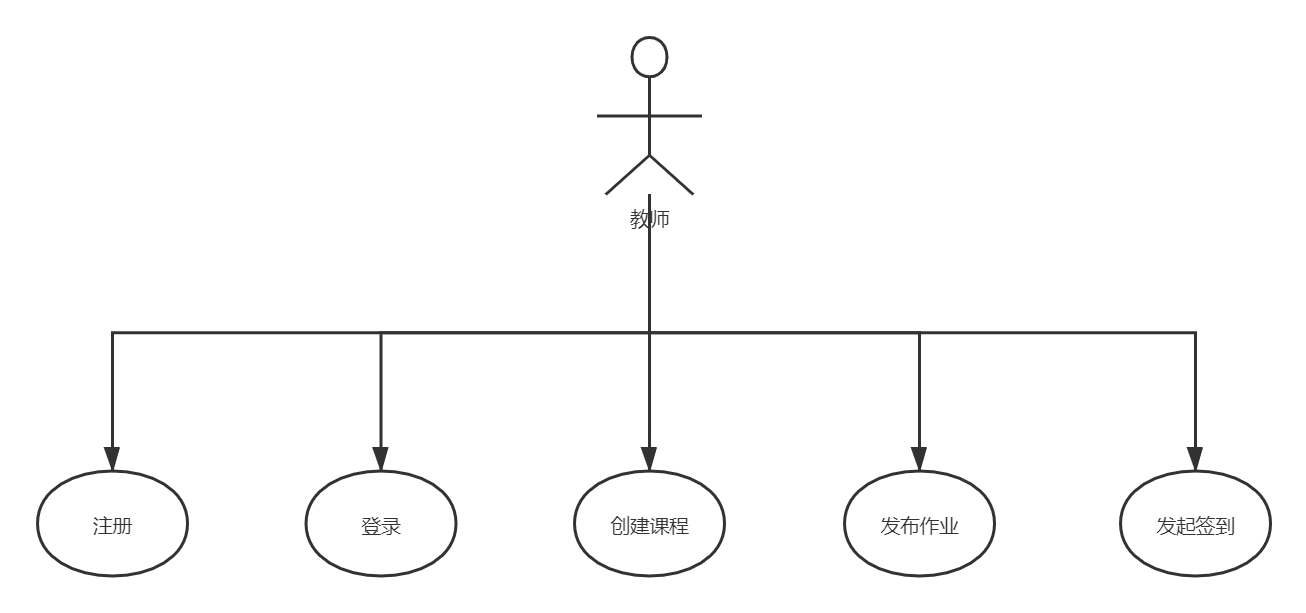
产品信息结构图如下：



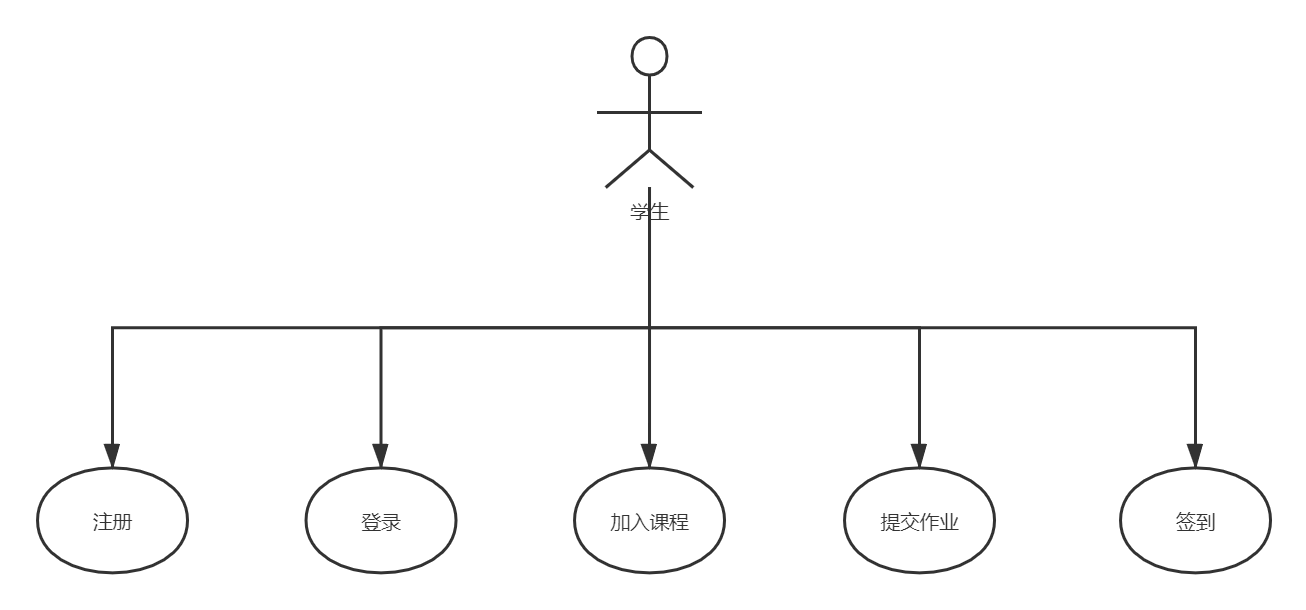
**2.4 用户**

本产品用户包括高校老师以及学生两类，其主要特点如下两部分用例图所示：

教师：

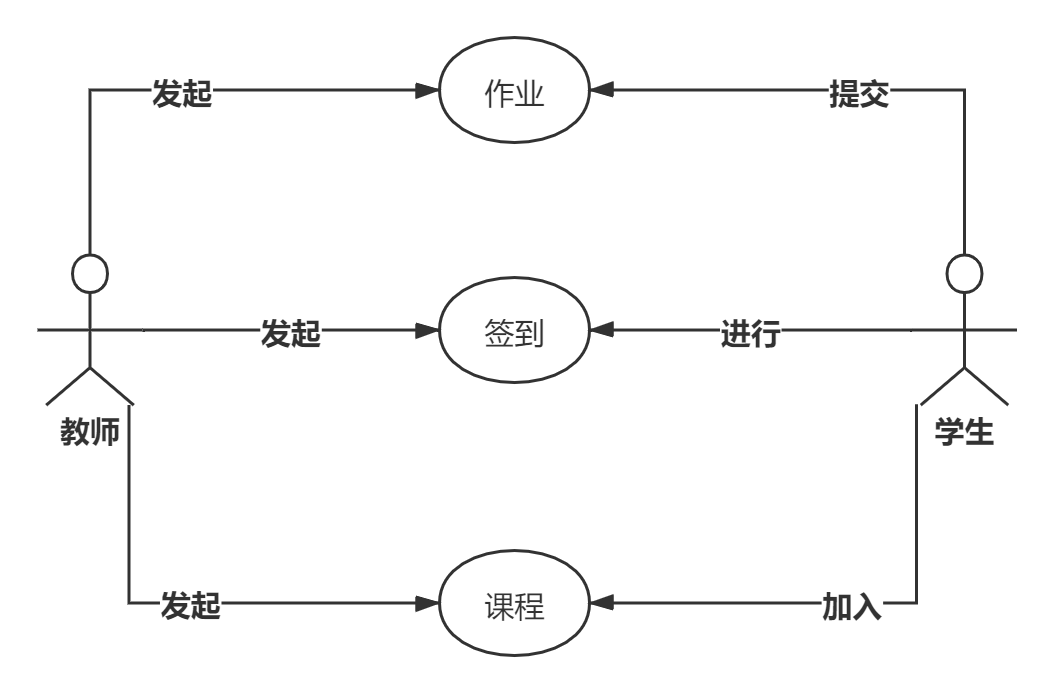


学生：



**2.5 业务流程**

其中整体流程图如下图所示：



**3 业务功能**

**3.1 注册与登录**

**3.1.1注册与登录**

需求描述：教师、学生在未登陆或注册的情况下，通过该功能实现APP的注册与登录。

业务场景：小黄打开APP后，未注册的情况先选择手机号注册，填写资料并填写收到的验证码进行注册，注册后进行登录，登录可选择手机验证码或密码登录。

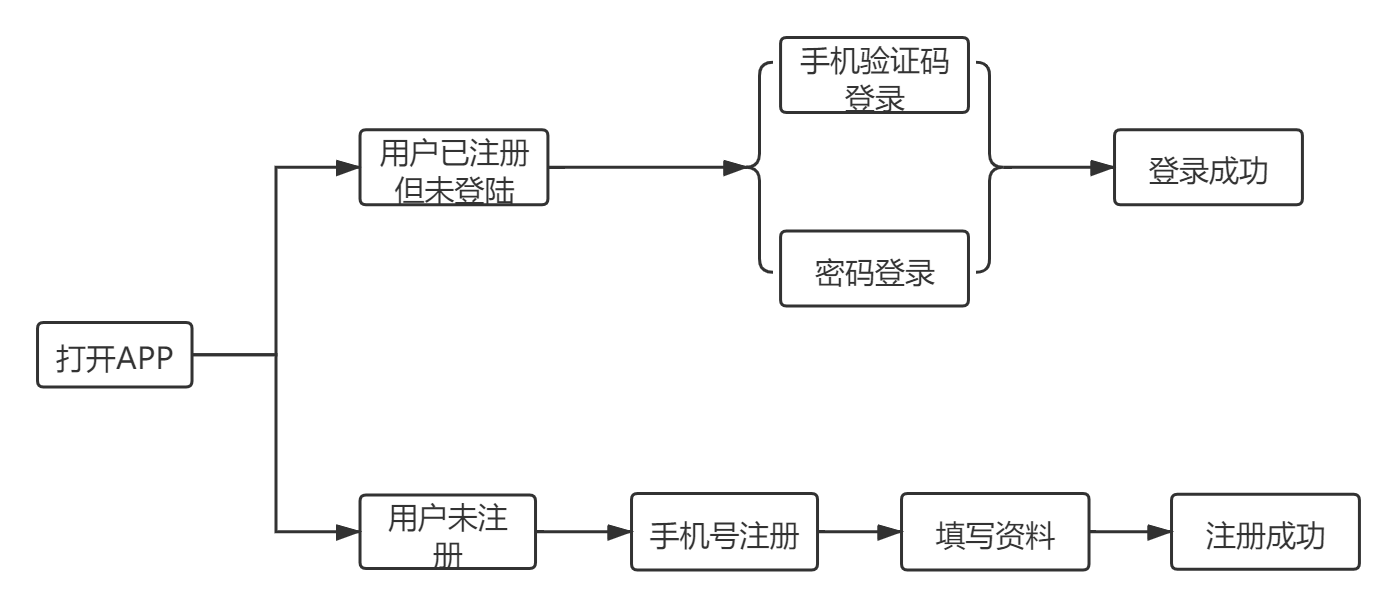
参与者：教师、学生

前置条件：打开APP

触发条件：打开APP但未登录

基本事件流：1、打开APP；2、未注册的用户选择手机号注册，输入手机号码，获取验证码，并填写资料完成注册；3、已注册的用户未登陆选择手机验证码或密码登录。

流程图：



后置条件:登录成功后进入主页面，并且下次打开APP无需再次登录。

**3.1.1.1 界面原型**

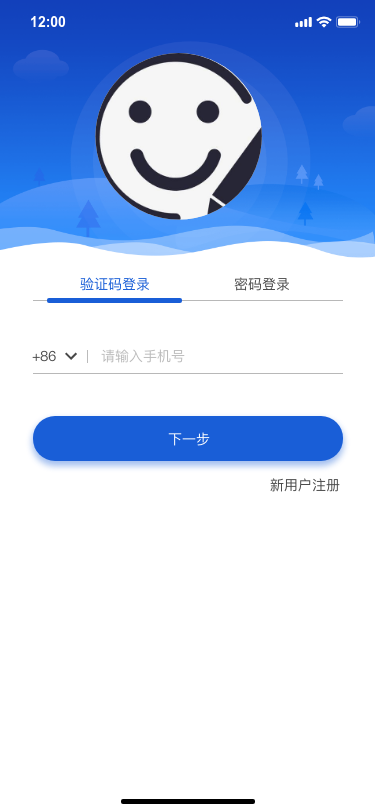


图1：验证码登录



图2：验证码登录-填写验证码

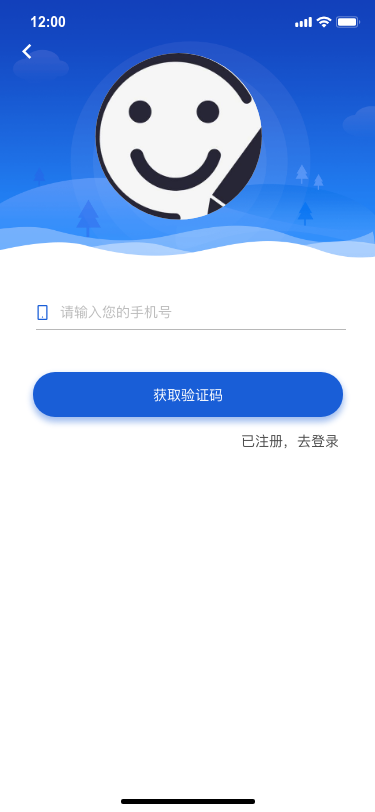


图3：验证码登录-获取验证码



图4：密码登录

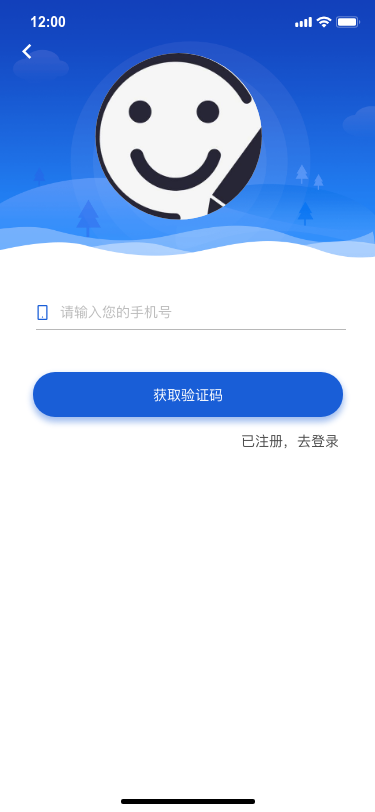
****

图5：注册获取验证码



图6：注册

**3.1.1.2 验收标准**

如果注册成功，保证能用注册信息正常登录。

**3.2 班课**

**3.2.1 创建班课**

需求描述：教师在首页面进行班级的创建，且只有老师有权限进行班级的创建

业务场景：黄老师点击首页面右上角“＋”符号，选择创建班级，填写班级资料后创建成功，获得班级ID.

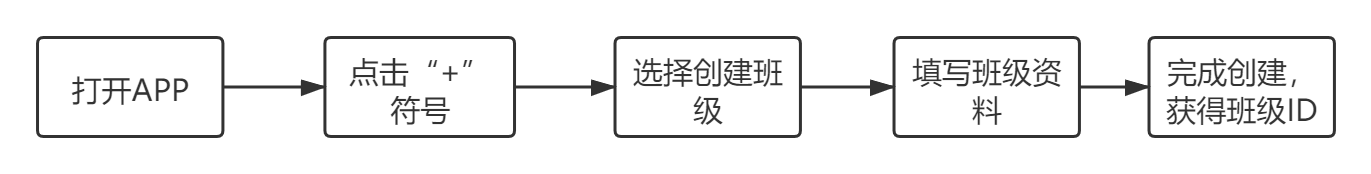
参与者：教师

前置条件：用户是教师并已经登录

触发条件：右上角“＋”符号，选择创建班级

基本事件流：1、打开APP；2、点击右上角“＋”符号；3、选择创建班级；4、填写班级资料；5、创建完成，获得班级ID。

流程图：



后置条件:创建成功后，在“我创建的班级”里显示此班级。

**3.2.1.1 界面原型**



图7：选择创建班课



图8：创建班课-首页



图9：创建班课-信息填写

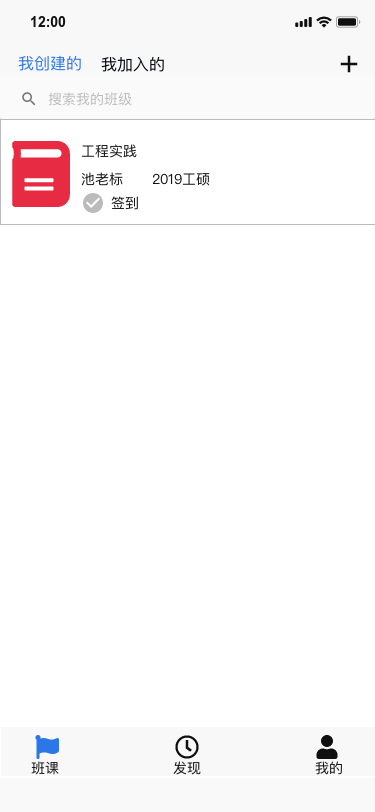


图10：创建班课-创建完成

**3.2.1.2 验收标准**

如果创建成功，能在“我创建的班级”里找到此班级，并且可以对其进行班级信息的编辑。

**3.2.2 加入班级**

需求描述：学生在首页面进行加入班级的操作

业务场景：黄同学点击首页面右上角“＋”符号，选择加入班级，填写班级ID后加入。

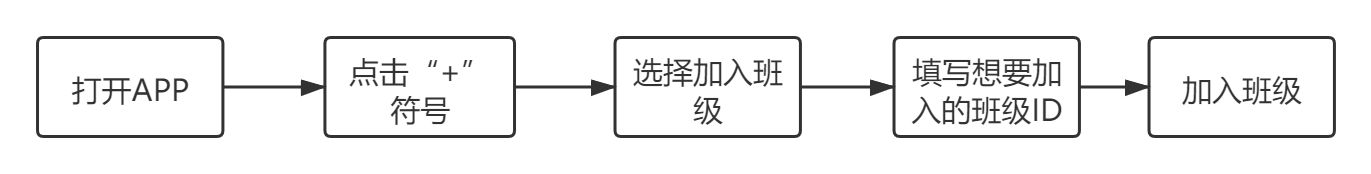
参与者：学生

前置条件：用户是学生并已经登录

触发条件：右上角“＋”符号，选择加入班级

基本事件流：1、打开APP；2、点击右上角“＋”符号；3、选择加入班级；4、填写想要加入的班级ID；5、加入班级

流程图：



后置条件:加入成功后，在“我加入的班级”里显示此班级。

**3.2.2.1 界面原型**



图11：加入班课-填写ID

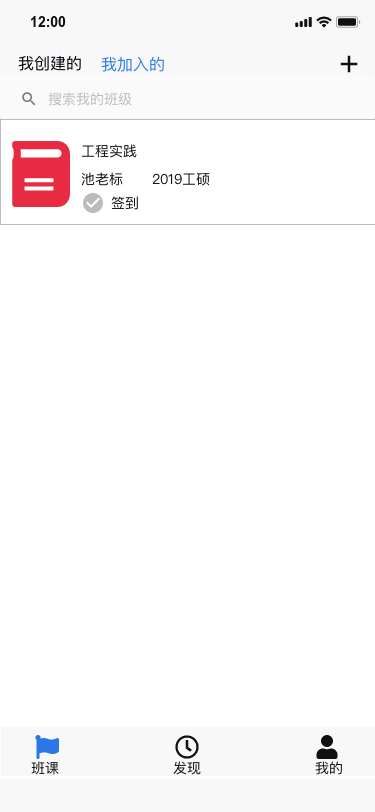


图12：加入班课-加入后界面

**3.2.2.2 验收标准**

如果加入成功，能在“我加入的班级”里找到此班级，并且可以点击进入此班级进行下一步操作。

**3.2.3 查看班级**

需求描述：学生用户点击“班课”页面的“我加入的班级”，可以查看已加入的班课的信息；教师用户点击“班课”页面的“我创建的班级”可以查看已创建的班级信息。

业务场景：黄同学点击“班课”页面的“我加入的班级”，查看自己已加入的班课的信息；黄老师用户点击“班课”页面的“我创建的班级”，查看自己已创建的班级信息。

参与者：学生，教师

前置条件：用户已经登录

触发条件：无

基本事件流：1、打开APP；2、在班课频道，教师点击我创建的、学生用户点击我加入的；3、分别查看到加入或创建的班级信息。

后置条件:无

**3.2.3.1 界面原型**

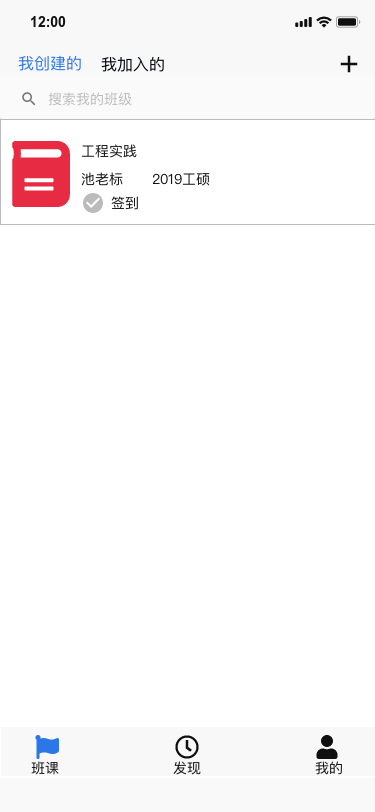


图13：查看班级-教师

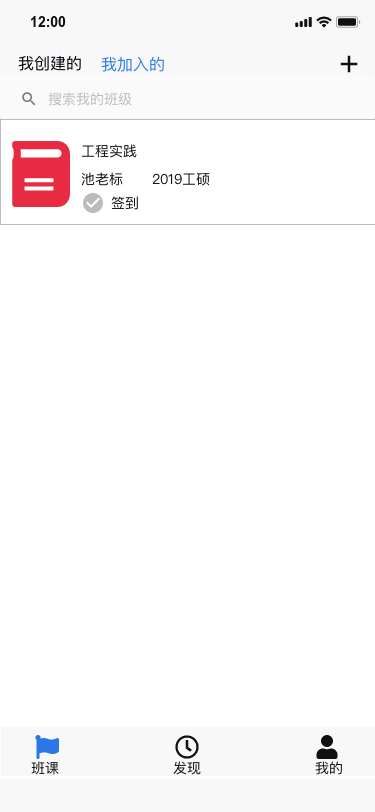


图14：查看班级-学生

**3.2.3.2 验收标准**

能在页面中正常的显示出创建或加入的课程

**3.2.3 作业**

需求描述：教师在创建的班级里选择发布作业；学生在加入的班级里，选择发布的作业进行提交。

业务场景：黄老师在已经创建的班级里，选择发布作业，填写作业描述与要求后发布；黄同学在加入的班级里，看到有已经发布的作业，选择已发布的作业，填写作业答案或提交附件进行作业的提交。

参与者：教师，学生。

前置条件：用户已经登录。

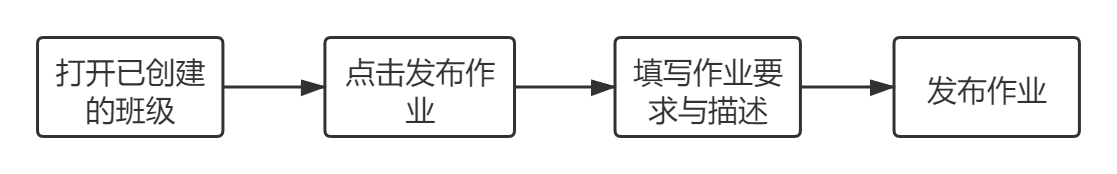
触发条件：进入已经创建或者加入的班级。

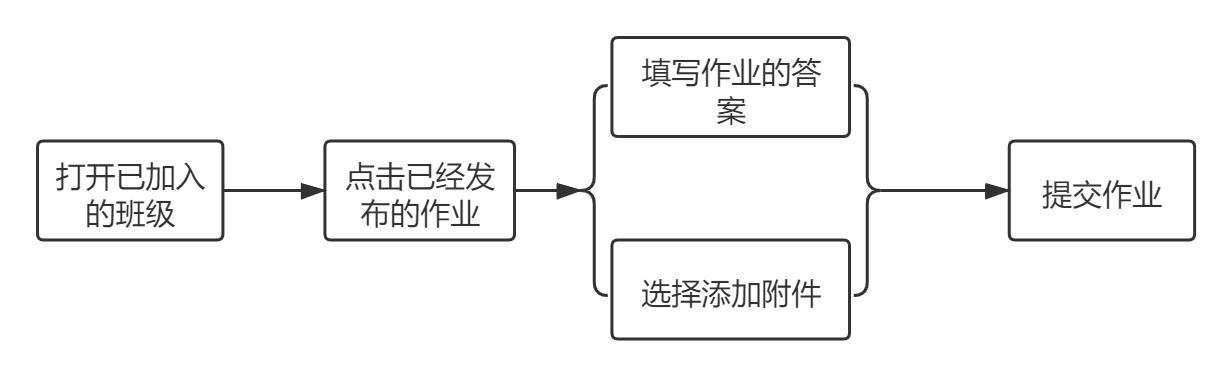
基本事件流：

教师：1、打开已创建的班级；2、选择发布作业；3、填写作业要求与描述；4、发布作业

学生：1、打开已加入的班级；2、点击已经发布的作业；3、填写作业的答案或者选择添加附件；4、提交作业。

流程图：





后置条件：教师发布作业成功后，在我的班级里显示此作业；学生提交成功后，提示提交成功，并在学生页面中，该作业上显示已提交。

**3.2.2.1 界面原型**

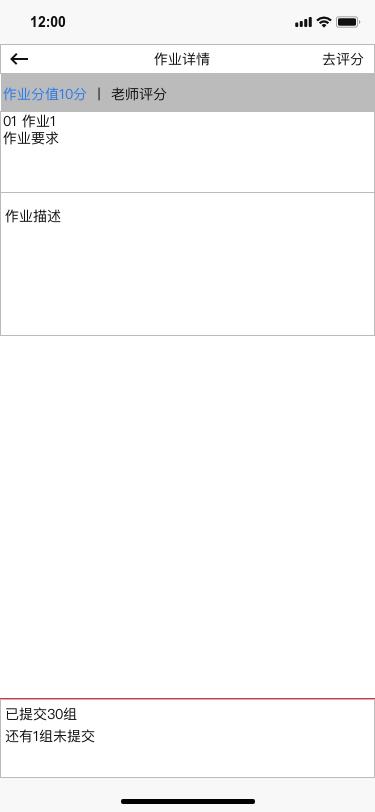
****

图14：作业页面-教师

****

图15：作业页面-教师发布作业

****

图16：作业页面-教师查看提交班级

****

图17：学生/教师页面-查看作业

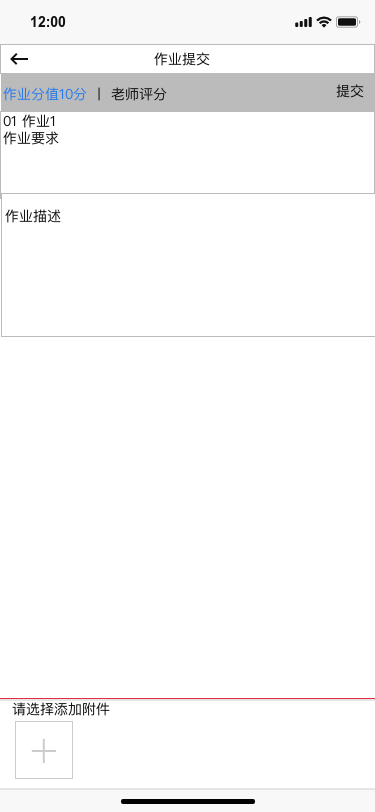
****

图18：学生提交作业

****

图19：学生提交成功提示

**3.2.2.2 验收标准**

如果加入成功，能在“我加入的班级”里找到此班级，并且可以点击进入此班级进行下一步操作。

**3.2.4 签到**

需求描述：老师在班级里进行签到，学生进行签到

业务场景：在班级，黄老师发布进行签到，学生进行签到直到老师停止签到。

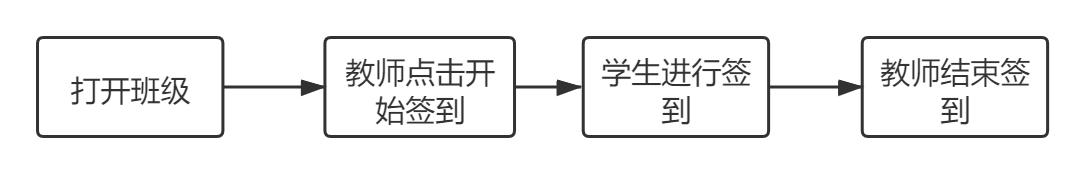
参与者：老师、学生。

前置条件：老师已经创建班级，学生已经加入班级。

触发条件：进入班级。

基本事件流：1、打开班级；2、教师选择发起签到；3、学生进行签到；4、教师结束签到。

流程图：



后置条件:教师端，提示开启签到与结束签到；学生端，提示签到成功。

**3.2.4.1 界面原型**



图20：教师发起签到



图21：教师发起签到-填写签到数字



图22：教师发起签到-结束签到



图23：学生签到界面



图24：学生签到-填写签到数字



图25：学生签到-签到成功

**3.2.4.2 验收标准**

学生在教师开始签到后，能成功进行签到。签到成功后，在本次签到中可以看到签到成功的提升，教师在开启和关闭签到时收到提示。

**3.2.5 搜索**

需求描述：老师在搜索框中搜索创建的班课；学长在搜索框中搜索加入的班级。

业务场景：黄老师在搜索框中搜索创建的班课；黄同学在搜索框中搜索加入的班级。

参与者：老师、学生。

前置条件：老师已经创建班级，学生已经加入班级。

触发条件：点击搜索框

基本事件流：1、点击搜索框；2、搜索

后置条件:将搜索结果显示在页面上，如果没有搜索结果则提升无结果。

**3.2.4.1 界面原型**

**无**

**3.2.4.2 验收标准**

能正确显示搜索结果，并且如果结果为无，则正确提示

**3.3 发现**

**3.3.1课程圈**

需求描述：教师、老师点击发现频道中的课程圈，能看到教师或者官方发出的信息分享。

业务场景：黄同学点击发现频道中的课程圈，能看到教师或者官方发出的信息分享。

参与者：教师、学生

前置条件：用户已登录

触发条件：点击课程圈

基本事件流：1、打开APP；2、点击发现；3、点击课程圈。

后置条件:无

**3.3.1.1 界面原型**

****

图26：发现首页

****

图27：课程圈页面

**3.3.1.2 验收标准**

正确显示课程圈内的内容

**3.4 我的**

**3.4.1 我的各功能**

需求描述：在“我的”一栏，我们可以查看“用户协议、设置、个人信息、关于软件”的信息；在设置中包括“关于我们、信息反馈、版本信息、退出登录”的操作；在个人信息中可以进行个人信息的编辑。

业务场景：如需求描述所示。

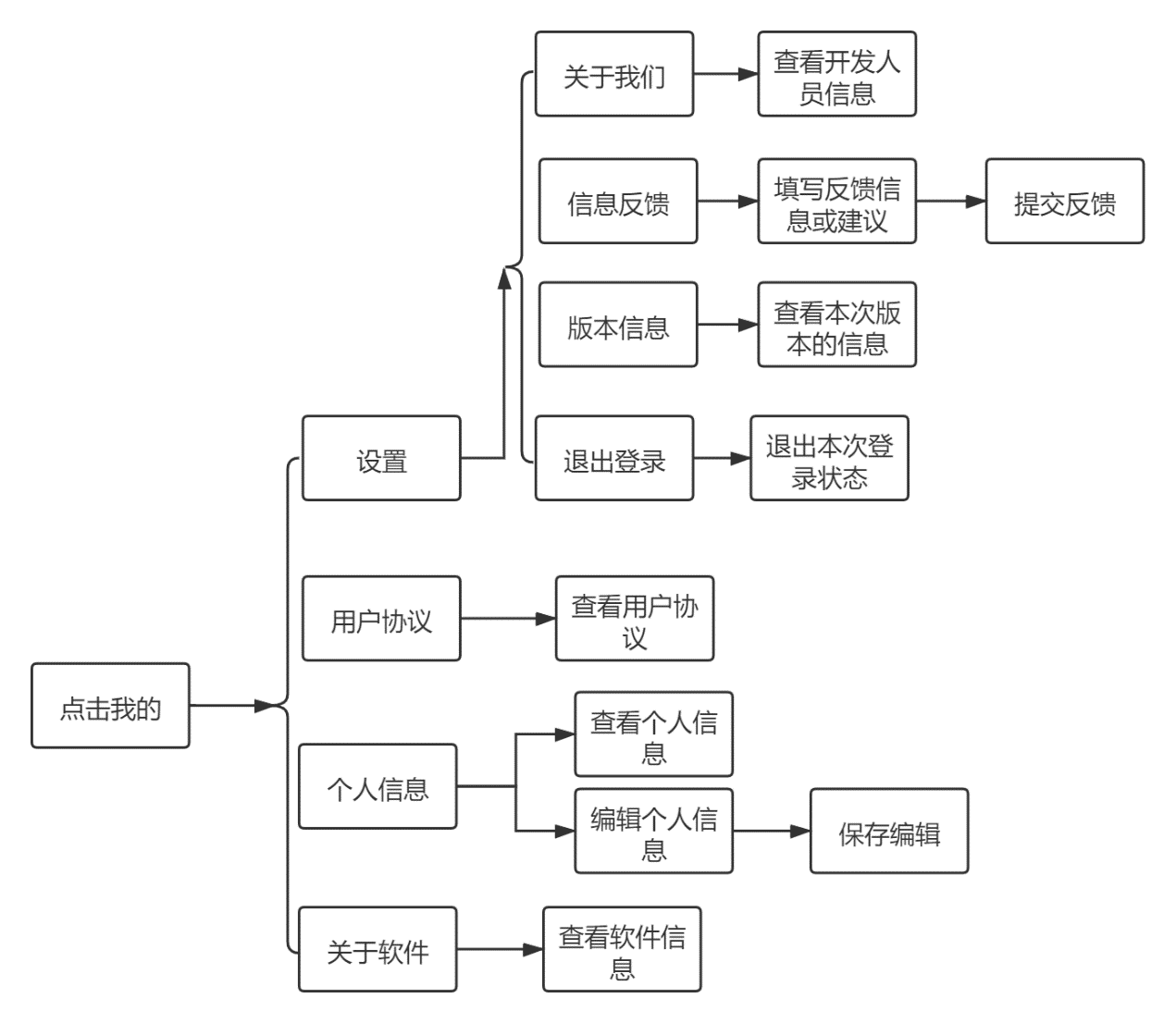
参与者：老师、学生。

前置条件：已经登录。

触发条件：在首页点击“我的”。

基本事件流：1、打开“我的”；2、选择用户协议可查看软件的用户协议；3、点击关于软件可查看软件的信息；4、个人信息，可查看个人信息并可以进行个人的信息的编辑；5、选择设置，可以看到“关于我们、信息反馈、版本信息、退出登录”的选项，其中“关于我们”包含开发人员信息；“信息反馈”可进行bug和其他建议的反馈；“版本信息”可以查看此软件该版本的信息；“退出登录”可退出此次登录状态。

流程图：



后置条件:1、信息反馈提交后，提示提交成功；2、编辑个人信息并保存后，提示编辑成功；3、退出登录，回到未登录状态。

**3.4.1.1 界面原型**

****

图28：我的首页



图29：我的—修改信息

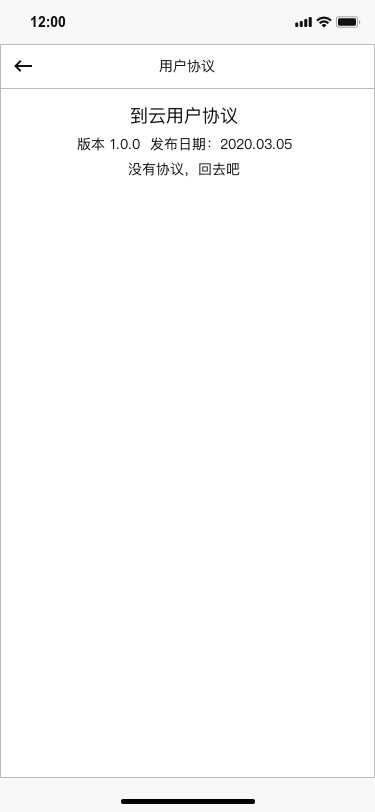


图30：我的—用户协议

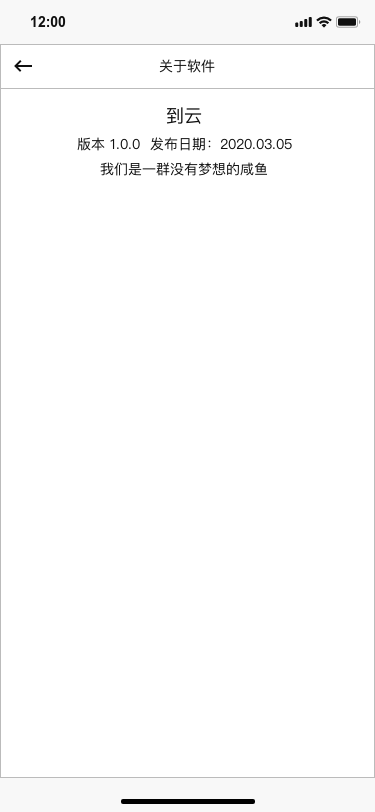


图31：我的—关于软件



图31：我的—设置

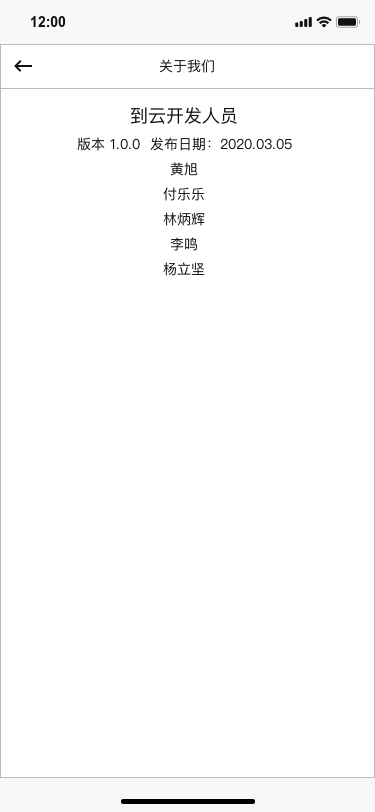


图32：我的—设置-关于我们



图33：我的—设置-信息反馈

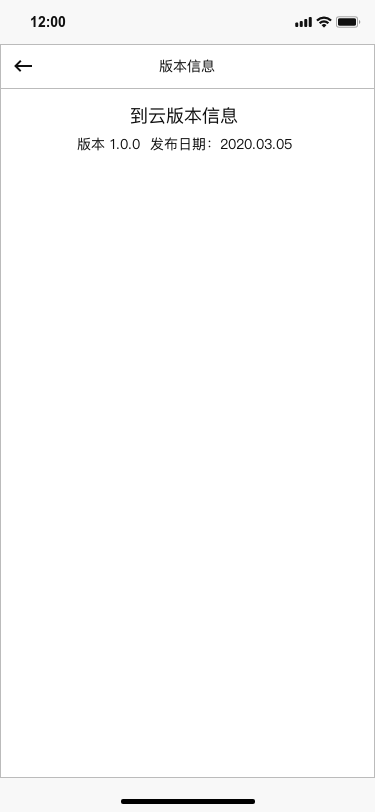


图34：我的—设置-版本信息

**3.4.1.2 验收标准**

1、信息反馈提交并提示提交成功后，后台保存该反馈信息并能查看；2、编辑个人信息保存后，在个人信息页显示已经更改后的信息；3、退出登录后能成功退到未登录状态。4、查看各种信息能正确显示。

**4 非功能需求**

1、用户界面设计要尽可能简洁大方，便于用户操作，整体界面的设计操作提示性要强，方便用户进行快速理解与功能操作。

2、系统的接口设计，数据传输要符合通用标准，避免因系统设计问题造成数据传输不畅，从而影响用户操作体验。

3、系统功能要分别清楚，不同功能要具体化为一个模块，既方便用户的操作，也方便开发人员的分工与后期维护。同时不同模块之间的协作要流畅，信息要互通有无。