<到云>后台开发需求说明书

小组成员：钱翊仁、许熠、王毅、汤博韬、魏振庭

目录

[1 引言 3](#_Toc3505)

[1.1 目的 3](#_Toc11573)

[1.2  背景 3](#_Toc13346)

[1.3 定义 3](#_Toc6875)

[2 项目概述 4](#_Toc12381)

[2.1 产品结构图 4](#_Toc7748)

[2.2 产品信息结构图 5](#_Toc26363)

[2.3 用户 5](#_Toc15487)

[2.4 业务流程 6](#_Toc14375)

[2.5 全局说明 7](#_Toc28418)

[3 功能详细需求 7](#_Toc12717)

[3.1 主页 7](#_Toc17896)

[3.1.1 个人信息 7](#_Toc32434)

[3.1.2 系统公告 8](#_Toc5463)

[3.1.3 密码修改 9](#_Toc27151)

[3.2 系统 10](#_Toc20194)

[3.2.1 用户权限 10](#_Toc12452)

[3.2.2 管理员管理 11](#_Toc28446)

[3.2.3 系统管理 13](#_Toc2845)

[3.2.4 公告发布 14](#_Toc26825)

[3.3 用户管理 15](#_Toc21245)

[3.3.1 教师、学生用户管理 15](#_Toc556)

[3.3.2 学生用户管理 17](#_Toc1262)

[3.4 课程管理 18](#_Toc32425)

[3.4.1 全部课程管理 18](#_Toc12967)

[3.4.2 我创建的课程管理 20](#_Toc32142)

[3.4.3 考勤管理 21](#_Toc18662)

[4 非功能需求 22](#_Toc164)

[4.1 界面友好 22](#_Toc29261)

[4.2 隐私安全 22](#_Toc24595)

[4.3 性能稳定 22](#_Toc26156)

**1 引言**

**1.1** **目的**

“到云”软件主要用于高校日常教学中的课堂管理，方便任课老师掌握和管理学生的情况，教学资料的分享以及课后电子版作业的收发。学生可通过“到云”软件参与课堂互动，与同学分享学习方法，提高课程参与度。

**1.2**  **背景**

“到云”，是由本小组合作开发项目，目前用户为高校师生。随着互联网的快速发展，当下网络课程的需求越来越大，同时配套线上教学的技术也越来越成熟，但是目前在高校的线上教学当中还是有相当的缺口。高校教学课程安排有相当大的弹性，非常适合线上教育教学的展开。而且面对高校出现教室资源紧缺，大批量学生上课让老师点名困难，管理考勤不方便等种种麻烦困难都可以考虑运用线上网络资源来解决。而我们小组开发的“到云”软件正是针对如今这样的缺口应运而生。

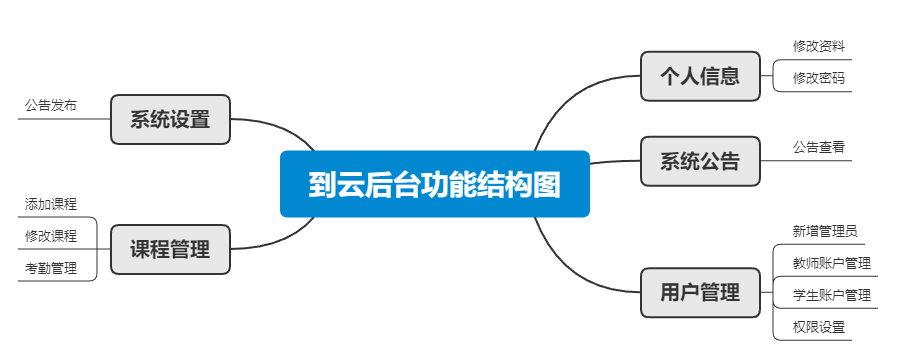
**1.3** **定义**

|  |  |
| --- | --- |
| **术语/缩略词** | **说明** |
| 前端Vue框架 | 一套用于构建用户界面的渐进式框架 |
| 前端Element-UI | 基于Vue2.0开发的一个组件库,提供了丰富的PC端组件。 |
| 后端Node.js | 基于Chrome V8引擎的JavaScript运行环境，使用了一个事件驱动、非阻塞式I/O 的模型。 |
| JavaScript | 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。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。 |
| MySQL | 最流行的关系型数据库管理系统，在 WEB 应用方面 MySQL 是最好的关系数据库管理系统应用软件之一。 |
|  |  |
|  |  |

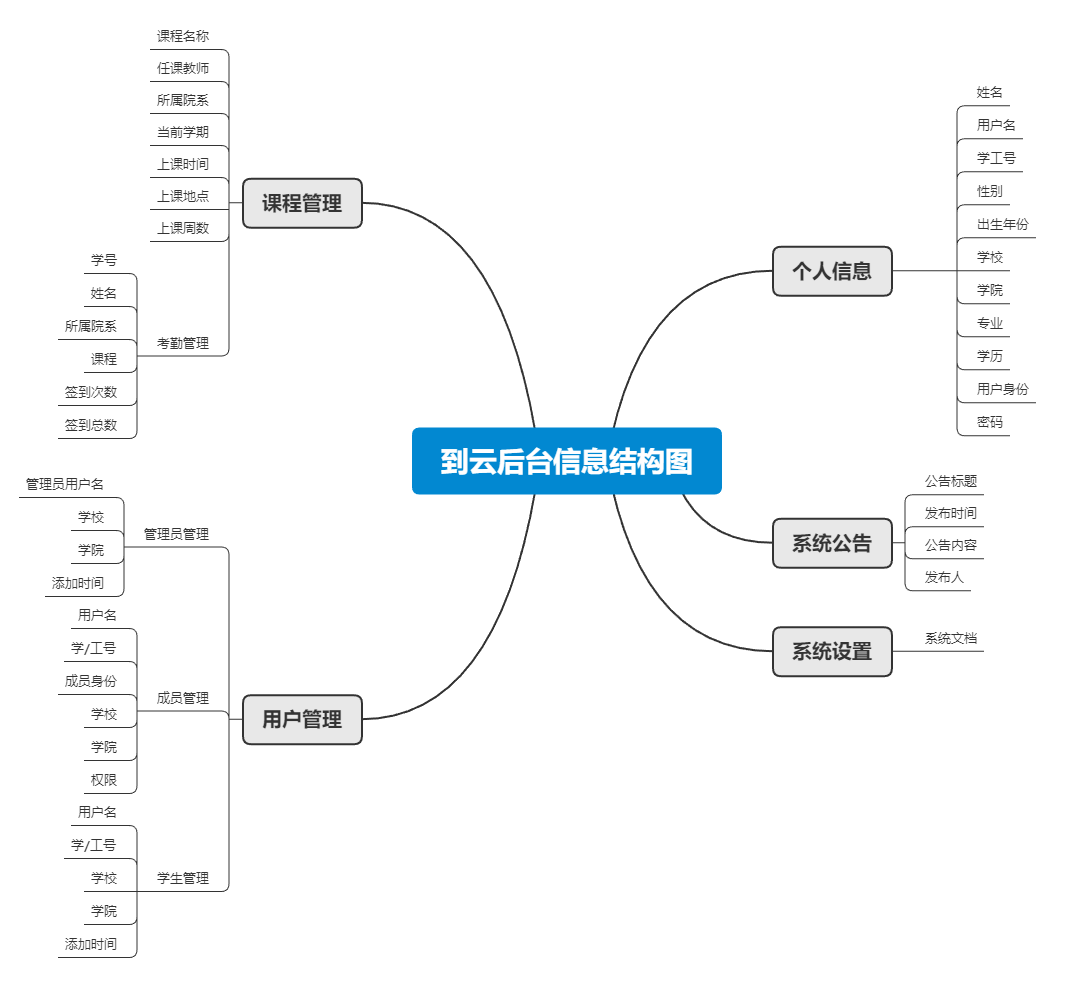
**2** **项目概述**

**2.1** **产品结构图**

|  |  |  |
| --- | --- | --- |
| **频道** | **页面** | **说明** |
| 个人信息 | 个人资料 | 查看、修改个人详细信息 |
| 密码修改 | 可以修改个人密码 |
| 系统公告 | 公告列表 | 查看所有公告 |
| 公告详情 | 查看公告详细内容 |
| 公告发布 | 发布新公告 |
| 用户管理 | 管理员管理 | 查看现有管理员 |
| 新增管理员 | 添加新管理员 |
| 教师/学生账户管理 | 管理员对教师或者学生的账户进行管理 |
| 学生账户管理 | 教师对学生的账户管理 |
| 权限设置 | 赋予教师/学生一定权限 |
| 新增成员 | 填写必须信息，添加新成员 |
| 课程管理 | 课程管理 | 查看现有课程 |
| 添加课程 | 可以添加新的课程 |
| 修改课程 | 可以对现有课程进行修改 |
| 系统设置 | 文档发布 | 查看现有系统文档 |
| 新增文档 | 可以编写新文档 |
| 登入注册 | 登录 | 后台系统登录界面 |
| 注册 | 后台系统注册界面 |
| 忘记密码 | 密码找回，修改界面 |

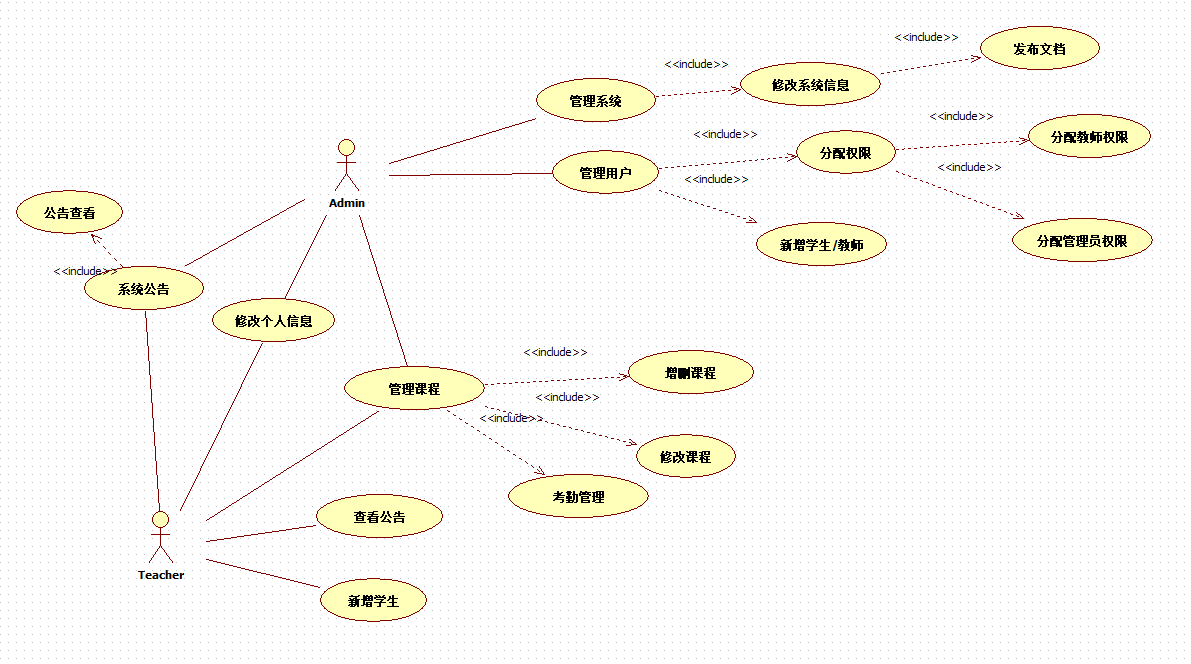
****

**2.2** **产品信息结构图**

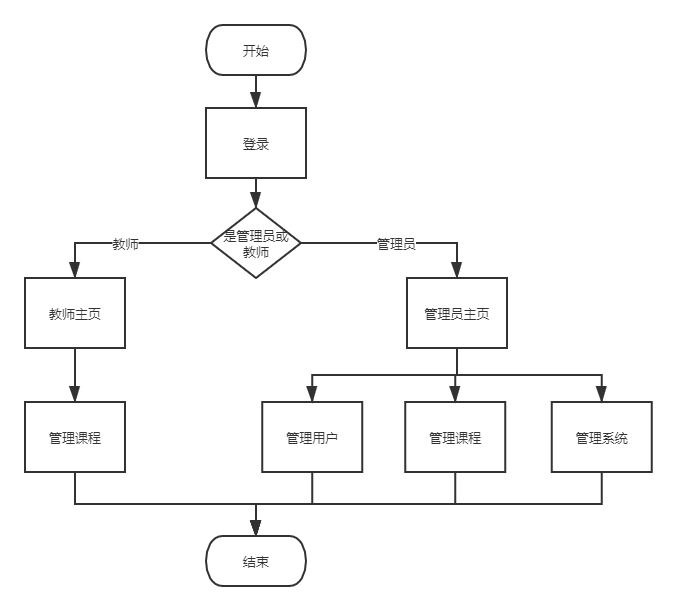
****

**2.3** **用户**

|  |  |  |
| --- | --- | --- |
| **角色** | **功能** | **特点** |
| 老师 | 新建课程、编辑课程、删除课程、添加学生、查看学生信息、删除学生、发布公告、发布签到 |  |
| 管理员 | 编辑系统信息、管理用户、管理课程 | 管理员具有最高权限 |



**2.4** **业务流程**

****

**2.5** **全局说明**

本系统中，所有用户都需要登录后才能进入系统进行操作。在登录页面有忘记密码选项，用户可以根据自己的预留信息修改密码。对于每个用户而言，都能修改自己的个人信息，例如修改昵称、更换头像等。点击“删除”时，需要有弹窗跳出，询问是否“确认删除？”。

**3** **功能详细需求**

**3.1** **主页**

**3.1.1 个人信息**

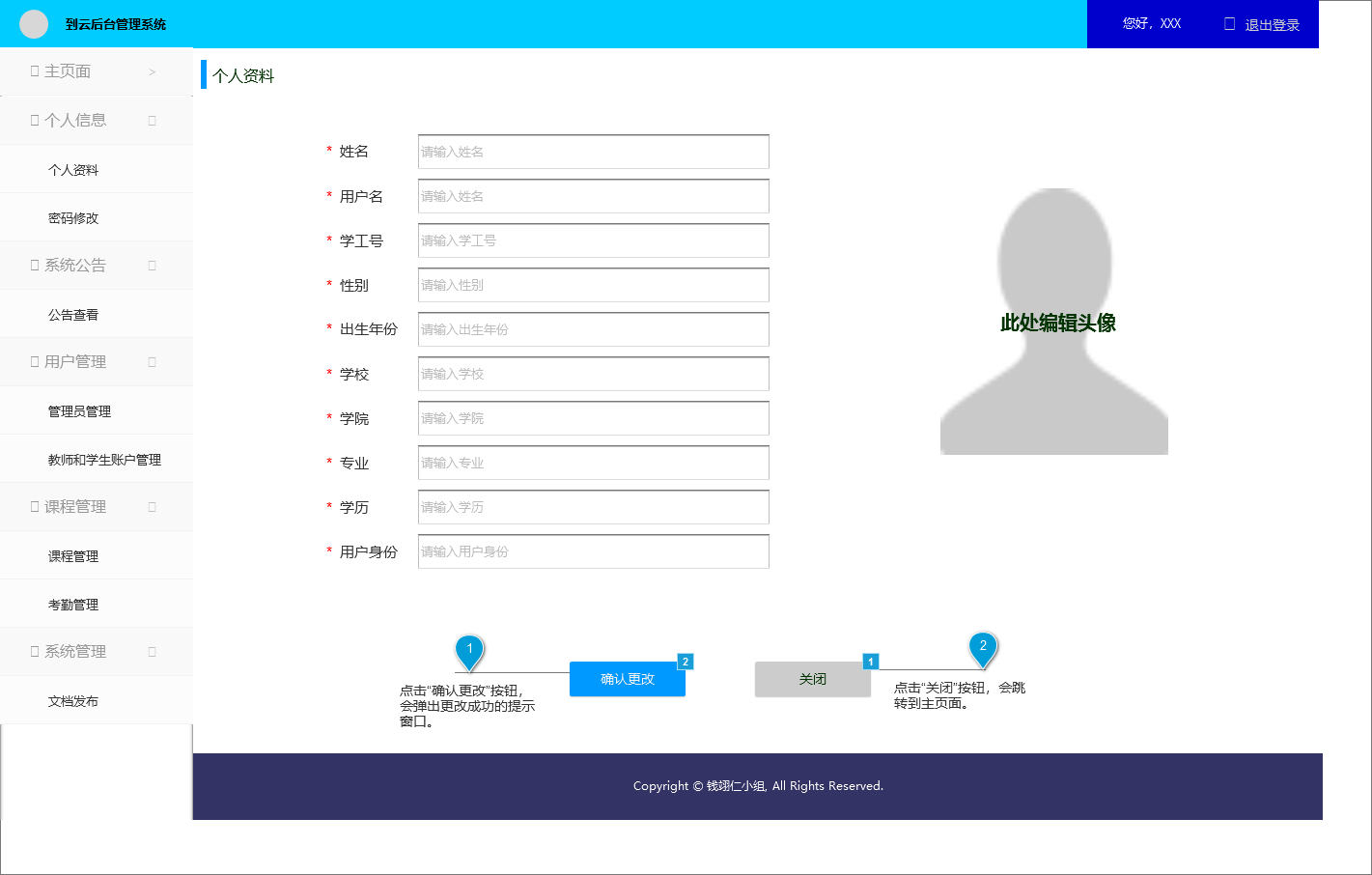
用户场景：从登入页面登入以后，再该页面可以看见该用户的各项个人资料。包含用户名，姓名，用户权限，头像，等等信息。也可以通过修改个人资料来进行修改，最底下一行还有“确认更改”和“取消”按钮来完成对个人资料修改的互动。

优先级：高

业务流程：无

输入/前置条件：老师或管理员用户登入

页面逻辑

   界面原型：

交互：用户可以通过点击相应的条目栏来进行编辑。可以对用户的姓名，头像等等的信息直接修改。用户编辑完成后，点击“确认更改”，就会更新数据库里面相应的资料。

字段：个人资料的数据来源于数据库，更改后又会直接更新数据库。

输出/后置条件：点击“确认更改”后，弹出"更改成功"小窗口。

点击“关闭”后，则不更改信息，直接跳出到主页。

版本：

**3.1.2 系统公告**

用户场景：在主页上的相应位置可以看见发布的系统公告

优先级：中

业务流程：无

输入/前置条件：管理员发布了系统公告

页面逻辑

  界面原型：

交互：用户可以点击相应的公告标题来进入具体的公告页面浏览信息

字段：管理员发布的公告直接存储与数据库中，其他用户所看见的公告来源于数据库。

输出/后置条件：点击“关闭”则跳出至主页。

版本：

**3.1.3 密码修改**

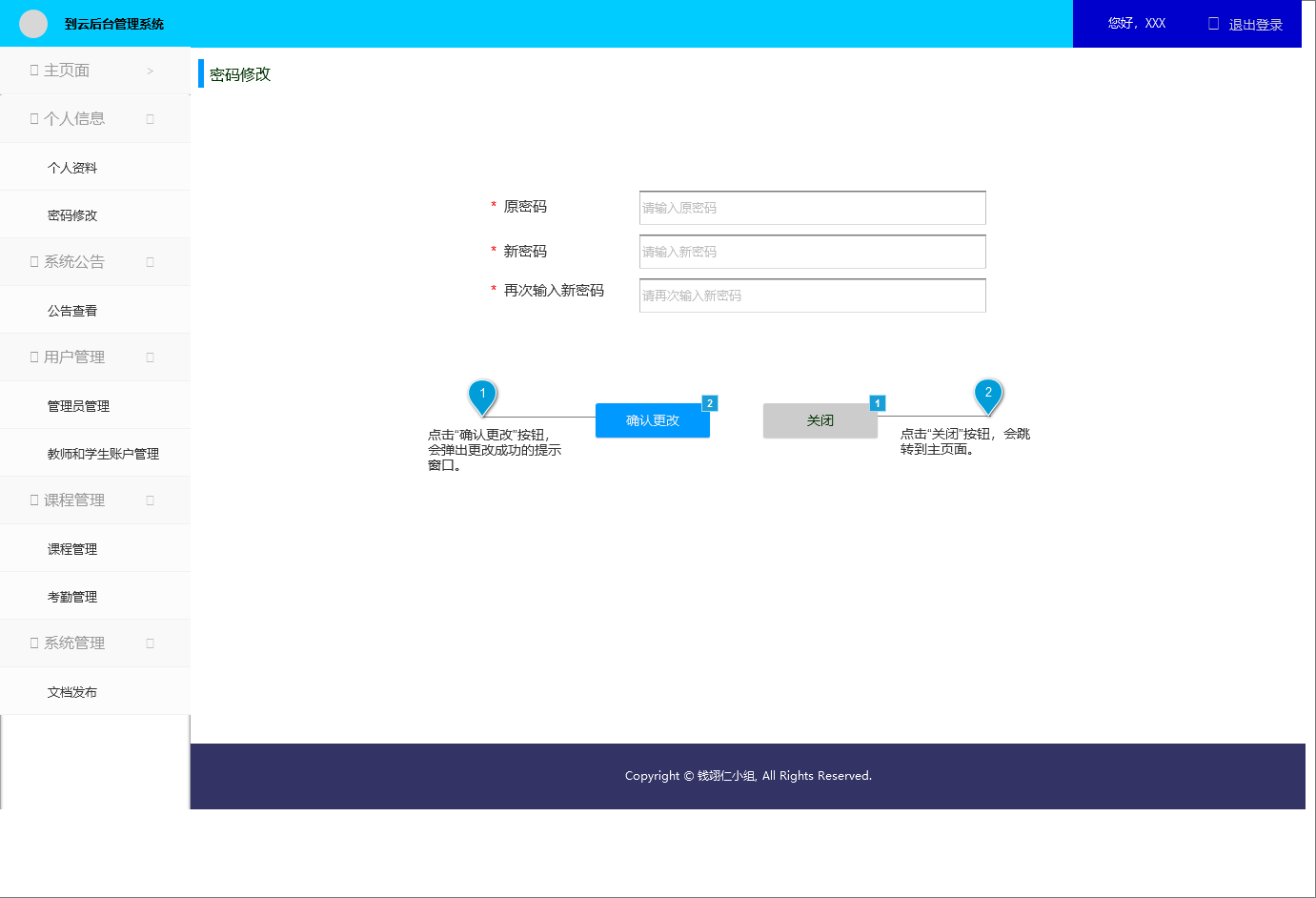
用户场景：在用户信息界面有一个"密码修改"的按钮，通过点击该按钮来变更用户自己的密码。其中分为三栏：原密码、新密码、再次输入新密码，通过在这三栏里面输入相应的信息来完成修改。

优先级：高

业务流程：无

输入/前置条件：进入用户信息页面

页面逻辑

  界面原型：

交互：用户在“原密码”、“新密码”、“再次输入新密码”这三栏中依次输入自己的信息来完成密码修改。

字段：用户密码必须要通过加密算法加密后再存储于数据库中

输出/后置条件：点击“确认更改”则跳出修改成功的小窗口；

点击“取消”则跳出至主页。

版本：

**3.2** **系统**

**3.2.1 用户权限**

用户场景：在管理员用户的主页上有一个“用户权限”的按钮，可以通过该功能来对每一个老师或者学生用户分配权限。比如老师用户的“创建班课”、“发起签到”的权限等等。

优先级：高

业务流程：无

输入/前置条件：管理员用户登入主页

页面逻辑

  界面原型：

交互：采用相应权限内容后的方框打勾方式，如果管理员打勾，表示为该用户授权该项权利；反之则没有。

字段：无

输出/后置条件：点击“提交”则跳出该变更成功的小窗口；

点击“取消”则跳出至主页。

版本：

**3.2.2 管理员管理**

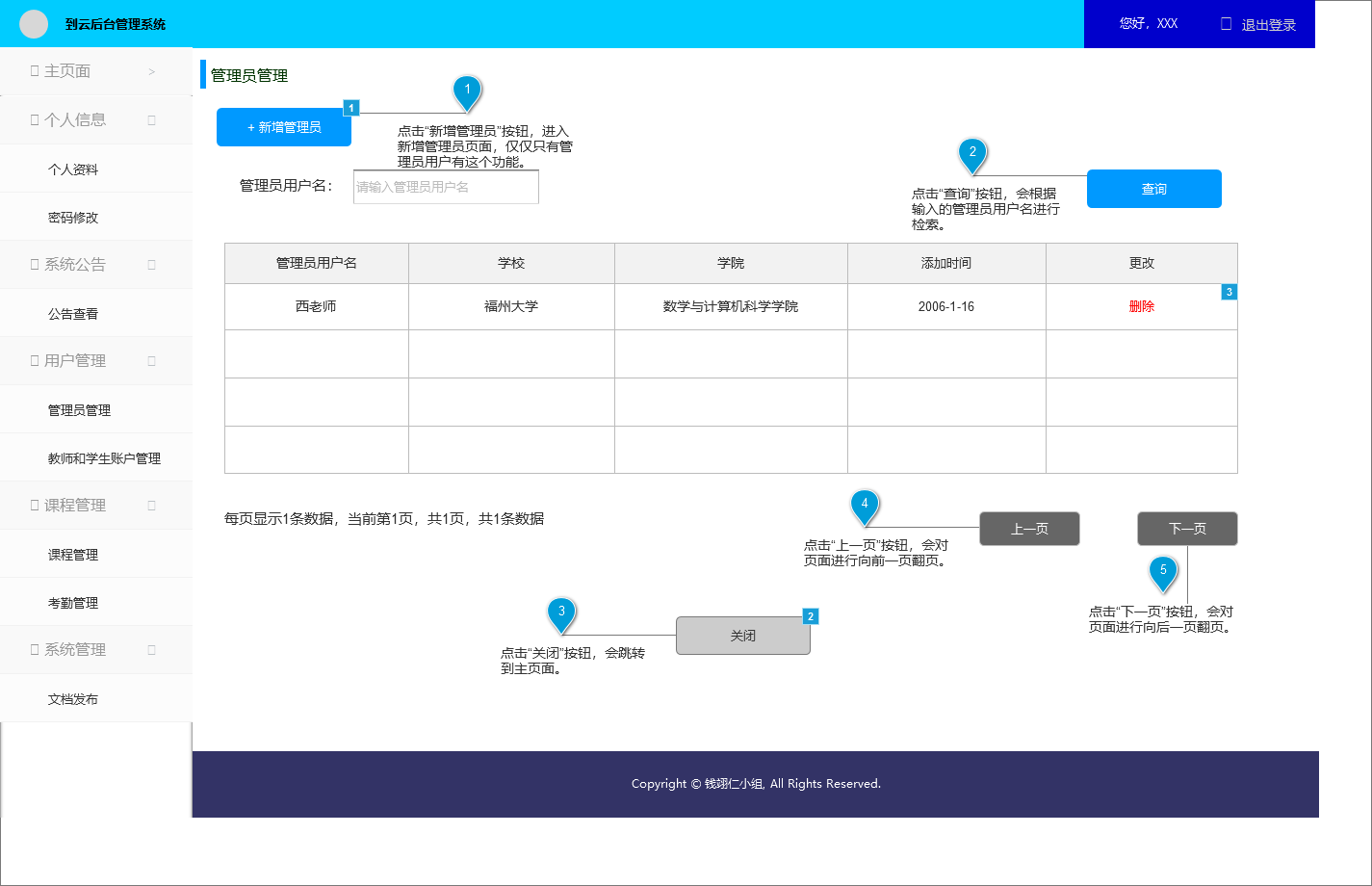
用户场景：在管理员用户的主页上有一个“管理员管理”的按钮，该功能可以新增或者删除管理员。

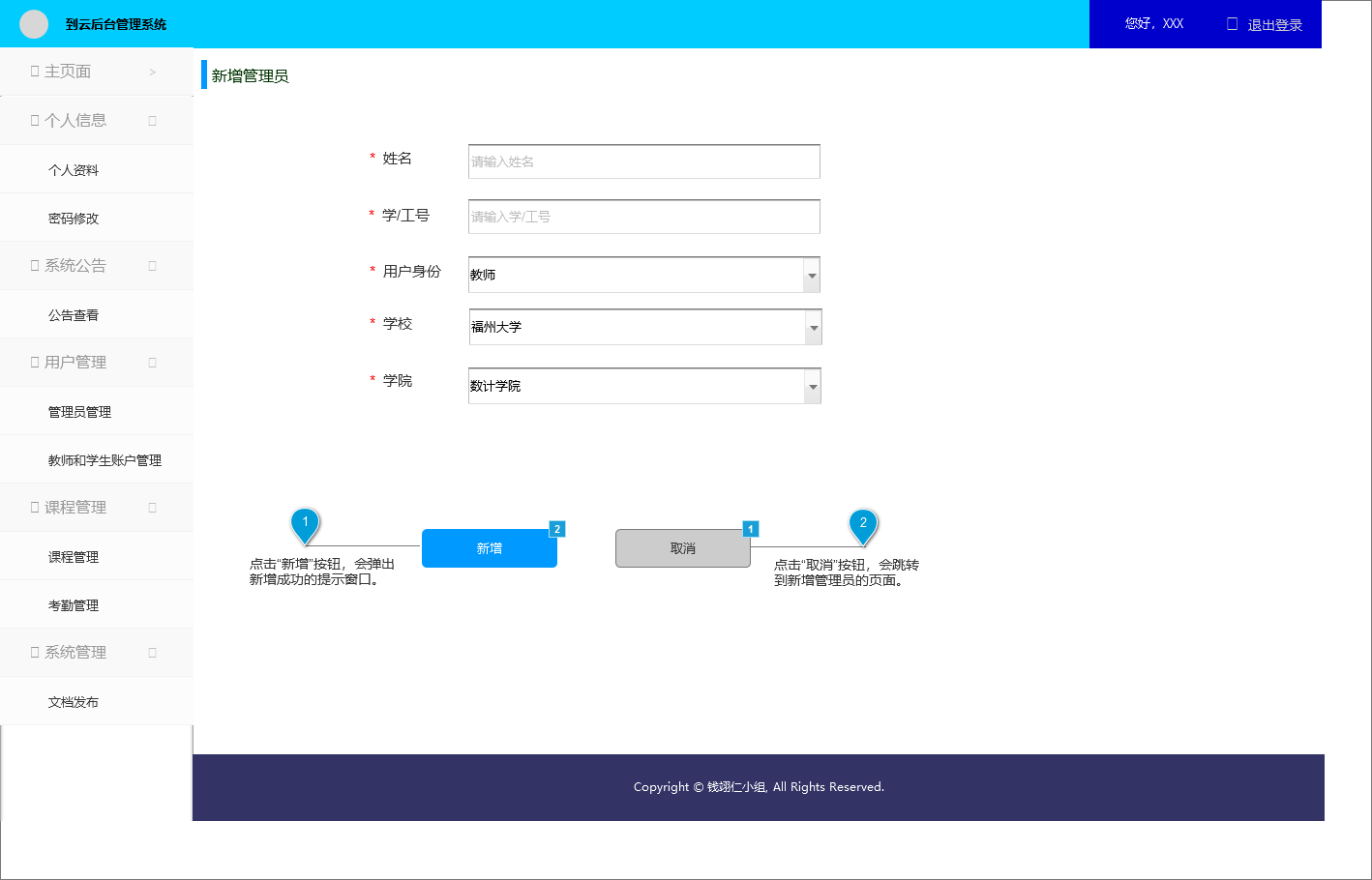
优先级：高

业务流程：无

输入/前置条件：管理员用户登入主页

页面逻辑

  界面原型：



交互：输入需要新增的管理员用户名，点击“新增”则授予该用户名管理员的权限。若点击“删除”，则该用户撤销管理员权限。但在撤销前会弹出“确认删除”窗口。

字段：新增的管理员会记录到数据库中，删除的管理员则从数据库中清除。

输出后置条件：点击“提交”，则弹出“更改成功”小窗口；

点击“取消”，则跳转至主页。

版本：

**3.2.3 系统管理**

用户场景：管理员可以通过该功能来对该产品进行一些文档性说明，比如“更新日志”、“关于我们”、“开源许可”等等。管理员发布以后会出现在每一个用户的APP上面。

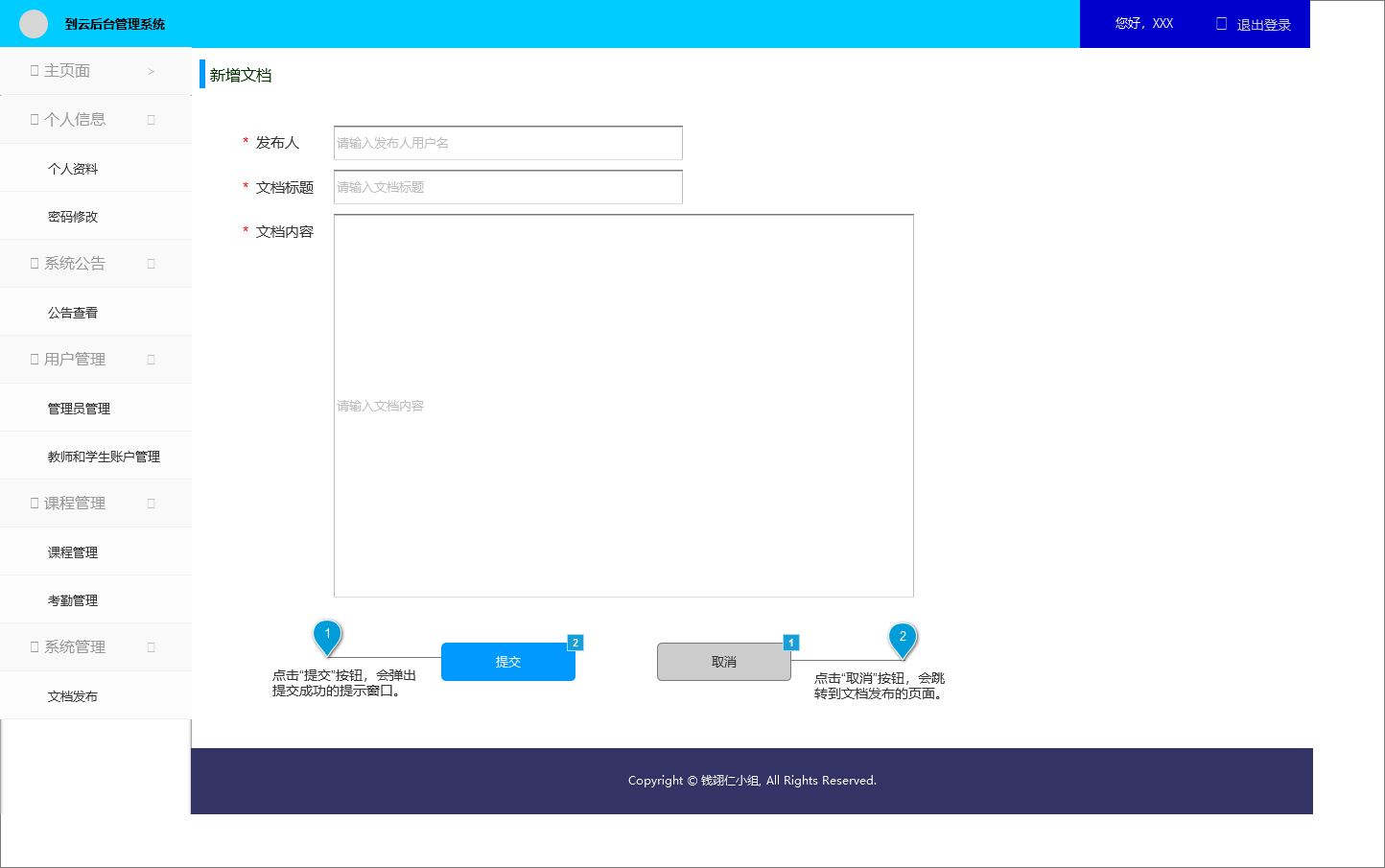
优先级：高

业务流程：无

输入/前置条件：管理员用户登入

页面逻辑

  界面原型：



交互：管理员可以新增删除文档，给文档取名相应的标题并添加内容到文档中，通过“提交”按钮同步到每一个用户的移动端上。

字段：这些文档内容都会同步在数据库上面

输出/后置条件：点击“提交”，则弹出“提交成功”小窗口；

点击“取消”，则跳转至主页。

版本：

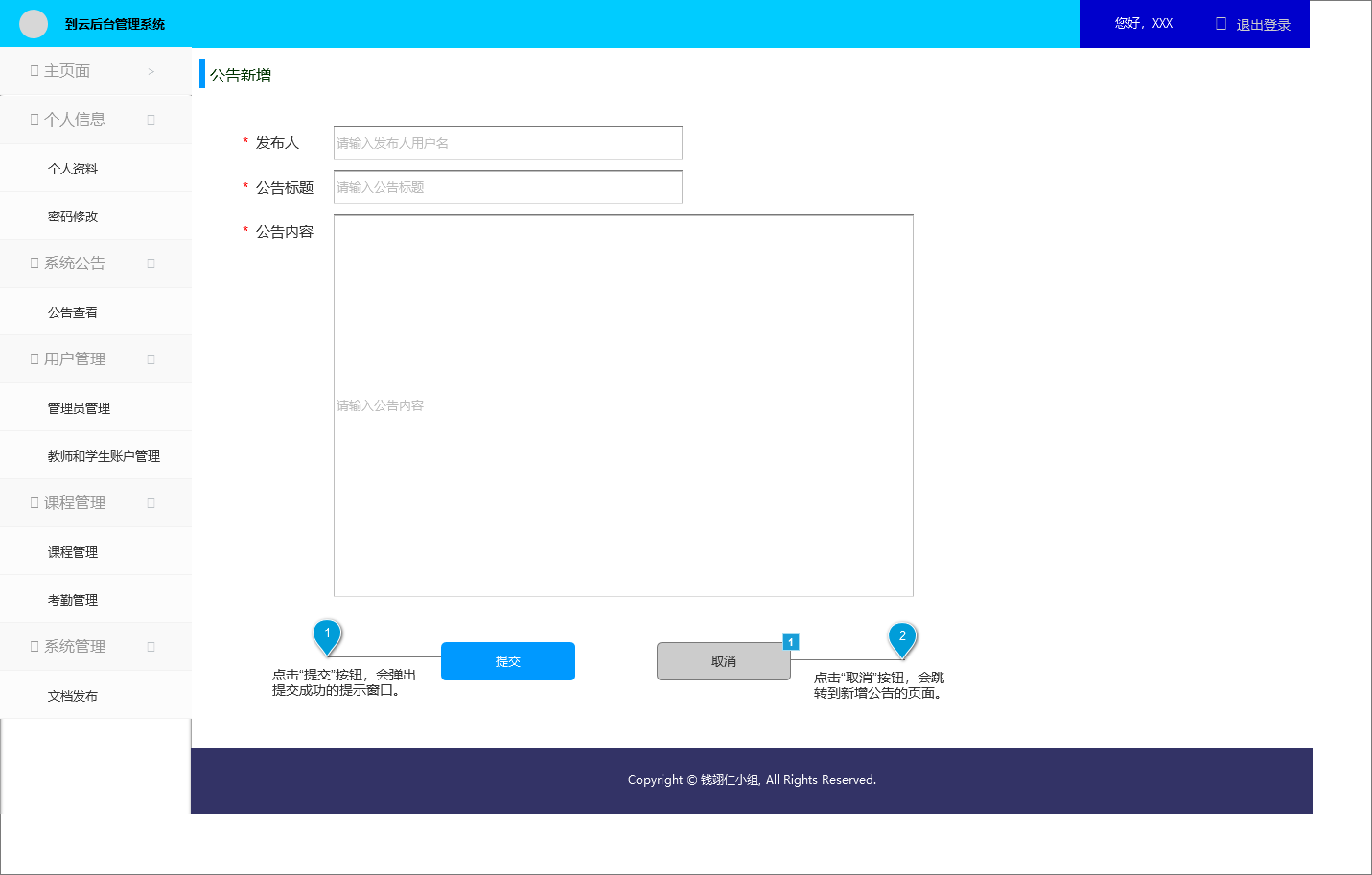
**3.2.4 公告发布**

用户场景：在管理员用户的主页上，有个"新增公告"的按钮，点击该按钮以后进入发布页面。在该页面中，包含公告的标题、内容。最下面一行有"提交"和"取消"两个按钮。

优先级：中

输入/前置条件：管理员用户登入

页面逻辑

界面原型：

交互：管理员用户可以在标题栏和内容框中编辑自己想要的内容

字段：管理员发布后的公告直接存储于数据库中

输出/后置条件：点击“提交”，则弹出“提交成功”小窗口；

点击“取消”，则跳转至主页。

版本：

**3.3** **用户管理**

**3.3.1 教师、学生用户管理**

用户场景：管理员点击“用户管理”进入此功能，可对教师、学生用户进行管理。

优先级： 高

业务流程：无

输入/前置条件：教师、学生用户已注册

页面逻辑

界面原型：

交互：管理员可以查看用户的基本信息；点击新增按钮后可以新增成员，点击删除按钮可以删除对应的用户。

字段：用户信息从数据库中读取；管理员若对数据进行操作会被保存到数据库中。

输出/后置条件：点击新增后进入新增成员界面，点击删除后返回用户列表并刷新。

版本：

**3.3.2 学生用户管理**

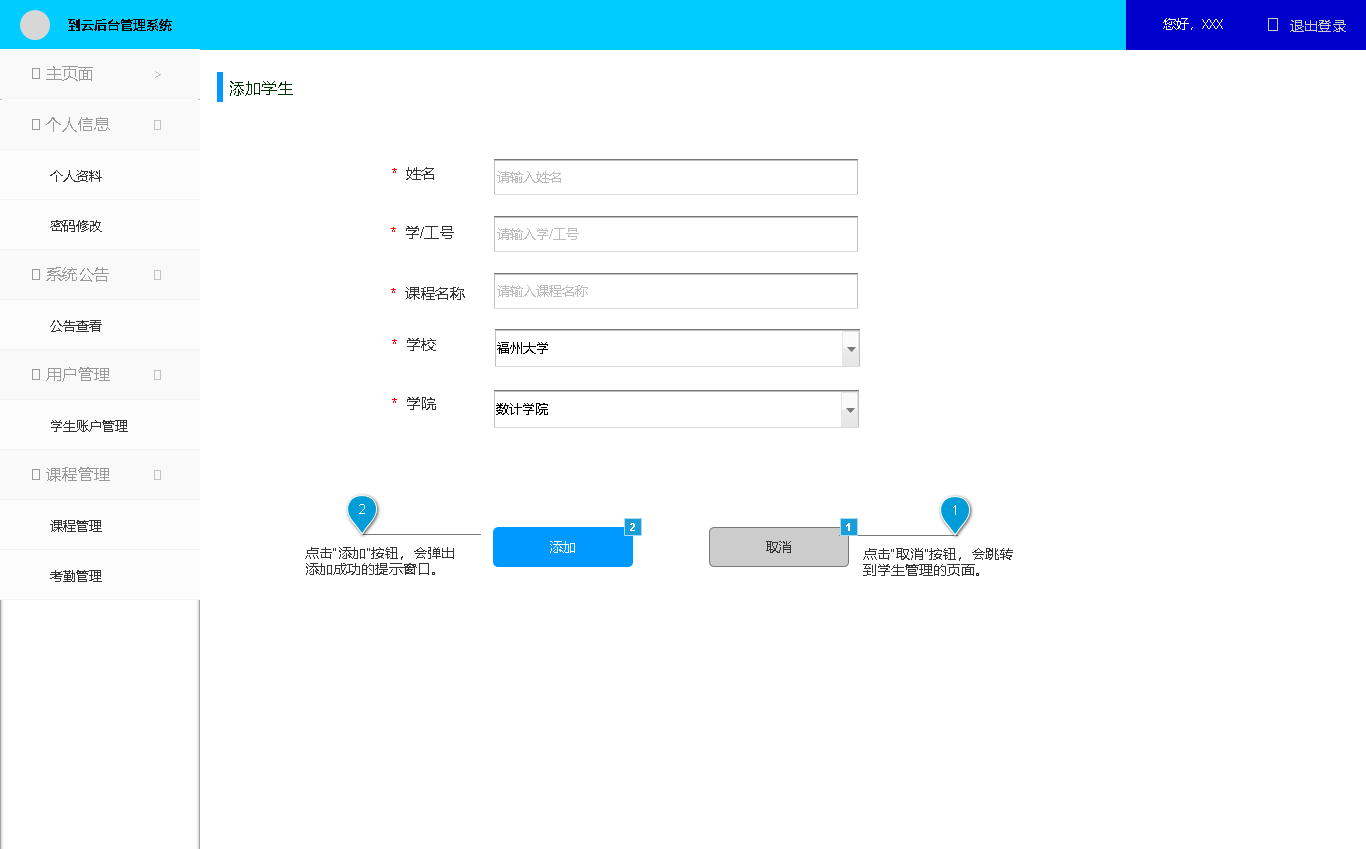
用户场景：教师点击“学生管理”进入此功能，可对学生用户进行管理。

优先级： 高

业务流程：无

输入/前置条件：学生用户已注册并加入教师创建的班课

页面逻辑

  界面原型：

交互：教师点击每个学生用户条目中的查看按钮可以查看学生用户的基本信息；点击删除按钮可以从班课中删除对应的学生用户，点击修改按钮可以修改学生用户的信息。

字段：用户信息从数据库中读取；

输出/后置条件：点击查看后进入学生用户信息界面，点击修改后进入修改学生用户信息界面，点击删除后返回学生用户列表并刷新。

版本：

**3.4** **课程管理**

**3.4.1 全部课程管理**

用户场景：管理员点击“课程管理”进入此功能，可对全部课程进行管理。

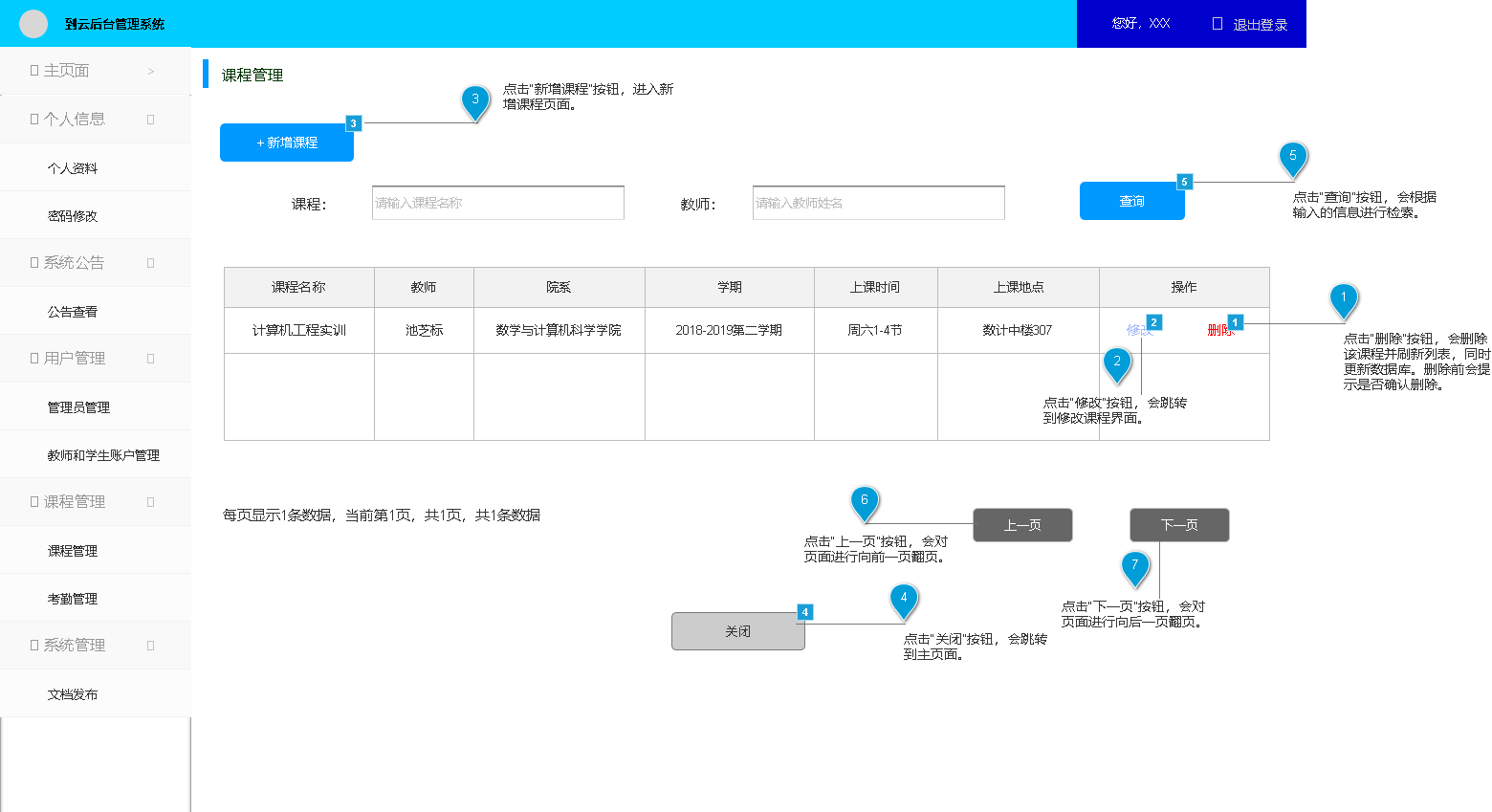
优先级： 高

业务流程：无

输入/前置条件：只能对已创建的课程进行查询、修改、删除操作

页面逻辑

界面原型：







交互：管理员可以查看课程的基本信息； 点击删除按钮可以删除对应的课程，点击修改按钮可以修改课程的信息。点击新增课程按钮可以创建新的课程。

字段：新增班课的名字长度为5-20个字节

输出/后置条件：点击修改后进入修改课程信息界面，点击删除后返回课程列表并刷新，点击新增课程按钮后进入创建课程界面。

版本：

**3.4.2 我创建的课程管理**

用户场景：教师点击“我创建的课程管理”进入此功能，可对该教师用户所创建的课程进行管理。

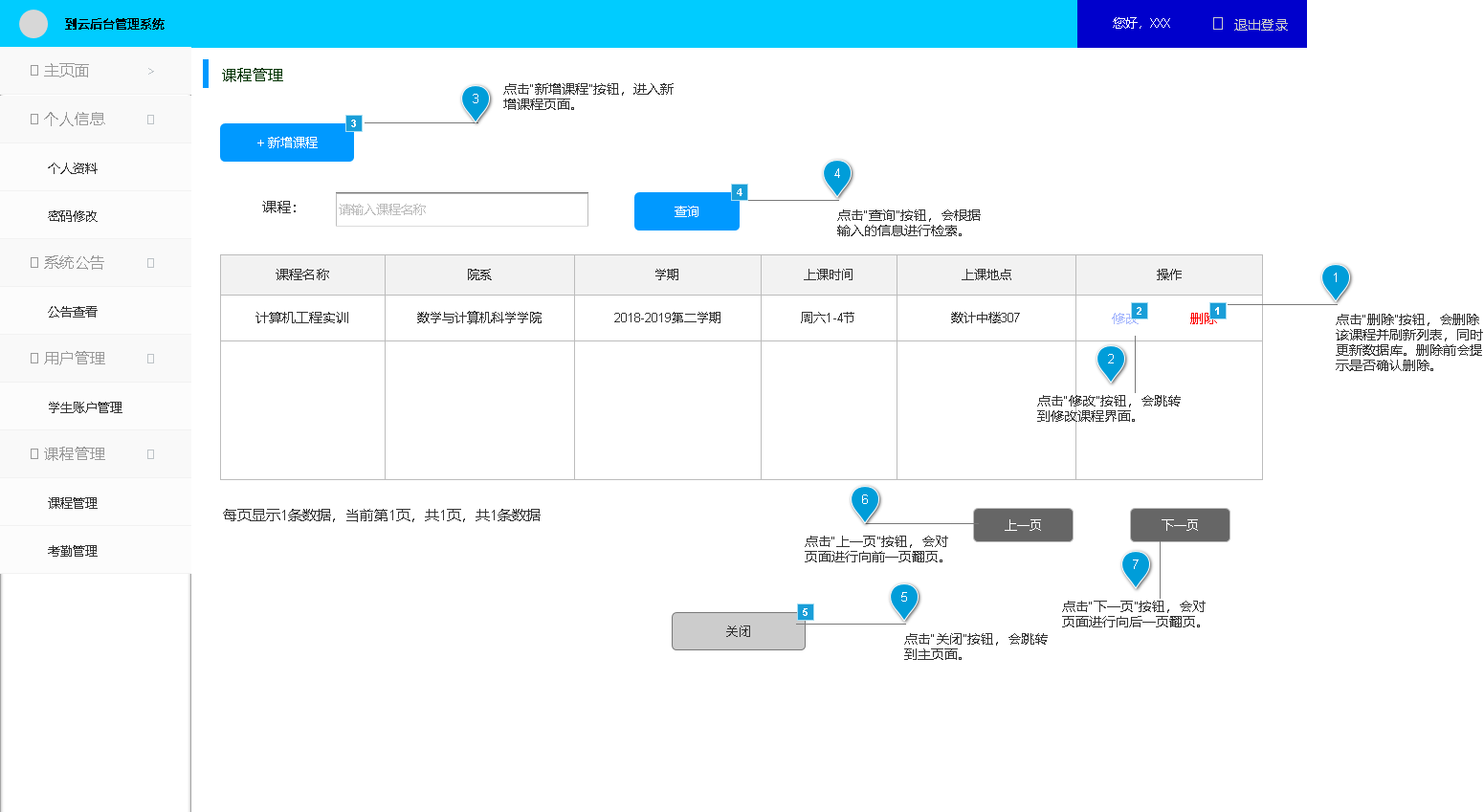
优先级： 高

业务流程：无

输入/前置条件：课程需要是教师用户所创建的

页面逻辑

  界面原型：







交互：教师可以查看课程的基本信息； 点击删除按钮可以删除对应的课程，点击修改按钮可以修改课程的信息。点击新增班课按钮可以创建新的课程。

字段：用户信息从数据库中读取；管理员若对数据进行操作会被保存到数据库中。

输出/后置条件：点击修改后进入修改课程信息界面，点击删除后返回课程列表并刷新，点击新增课程按钮后进入创建课程界面。

版本：

**3.4.3 考勤管理**

用户场景：教师点击“考勤管理”进入此功能，可以查看当前课程学生用户的考勤情况。

优先级： 高

业务流程：无

输入/前置条件：教师只能对自己创建的课程进行考勤情况的查看

页面逻辑

  界面原型：



交互：教师可以通过查询按钮查看对应学生的考勤情况，包括签到次数和总签到次数。

字段：考勤信息从数据库中读取；

输出/后置条件：无

版本：

**4 非功能需求**

**4.1** **界面友好**

开发的移动端程序的界面应该看过去简洁而舒适，让用户上手简单容易，一眼就明了各个图标的功能作用。

**4.2** **隐私安全**

该程序应保证用户的各项私人隐私数据不被泄露，保证后台管理系统的安全性。在注册、登入以及浏览信息的过程中不被他人窃取重要信息。

**4.3** **性能稳定**

程序的运行应该稳定不容易崩溃，而且运行应该流畅不容易卡顿，保证用户在体验本软件的流畅感。