

软件需求规格说明书

1、引言

1.1、目的

本文档将描述项目的各项功能需求、性能需求，阐述项目背景及范围等总体结构和功能结构，便于项目团队成员更好地了解项目情况，使项目工作开展的各个过程合理有序，也便于用户与开发成员进行沟通交流。

1.2、背景

随着现代科技的高速发展，学生的注意力被大大分散，逃课的现象日益严重，需要老师通过点名的方式进行监督。同时某些课程的容量太大，老师进行点名很耗时，期末时进行点名统计也十分麻烦。

1.3、 定义

文档中采用的专门术语的定义及缩略词简要如下：

术语	定义和信息
用户	所有注册该系统的用户。
成员	在一门课程中除了课程创建者外的所有用户。
课程	用户发起和完成签到的单位。
管理权限	加入课程的用户允许在该课程中发起签到、查看课程签到记录、导出签到结果。
已授权课程	课程创建者对用户加入某一课程后授予管理权限。
课程 ID	在用户创建一门课程后，系统自动对该课程生成唯一的标识码做为该课程的课程 ID。
Pin 码	由系统生成的 4-6 位的短数字码，用于判断签到的到场证明之一。
二维码	由系统生成的二维码，允许用户通过扫二维码加入课程或注册系统。
地理位置	通过获取移动端的地理定位信息来判断用户当前地理位置。
导出签到记录	允许用户以某种文件格式将签到记录导出到本地。

1.4、参考资料

[1] GB856T——88，软件设计文档国家标准—软件需求说明书

2、任务概述

2.1、目标

完成教务处的考勤要求，方便教师统计学生本学期课程出勤率，督促学生参与到课堂而开发了一个基于微信平台的签到系统。

我们期望打造出的产品有如下特性：

系统内教师和学生以课程为单位进行发起签到和完成签到以及对课程进行相应的管理。本签到系统需要教师和学生通过绑定微信并上传其真实姓名及学/工号成为注册用户，而实名信息由系统管理员维护。

在每门课程开始前，可以由用户创建课程，备注基本课程信息。在成功创建课程后，系统会自动生成唯一识别该课程的课程 ID 以及二维码。系统允许用户在自己创建的课程内进行删除课程和删除已加入课程的用户的操作，以及把课程管理权限分配加入课程的用户如助教、教务人员等，协同管理课程签到。

用户通过搜索相应的课程 ID 或扫描二维码加入该课程，如果课程结束或者误加课程，用户可以选择退出课程。

当需要签到时，具有管理权限的用户可以发起签到任务，并自定义签到时限和附加签到验证。若无任何验证，在签到时限内用户进入系统签到界面即可完成该次签到并显示签到成功。如果附加 PIN 码验证，发起签到的用户会收到系统随机生成标识该次签到的 PIN 码，在场的用户通过输入签到发起者公布的 PIN 码可完成 PIN 码验证。如果附加地理验证，加入课程用户在进入签到系统后，系统会尝试获取用户的地理位置，如果其地理位置与签到发起者的地理位置在合理范围内，则会满足地理验证。在签到时限内满足所有验证后，用户进入系统签到界面即可完成该次签到并显示签到成功。

具有管理权限的用户能够随时查看或导出所有已加入课程用户的签到记录，而加入课程的用户可以随时查看自己的历史签到记录。

本系统基于微信平台的小程序实现，采用 JavaScript 语言，主要面向移动平台。

2.2、功能需求

2.2.1、用例图

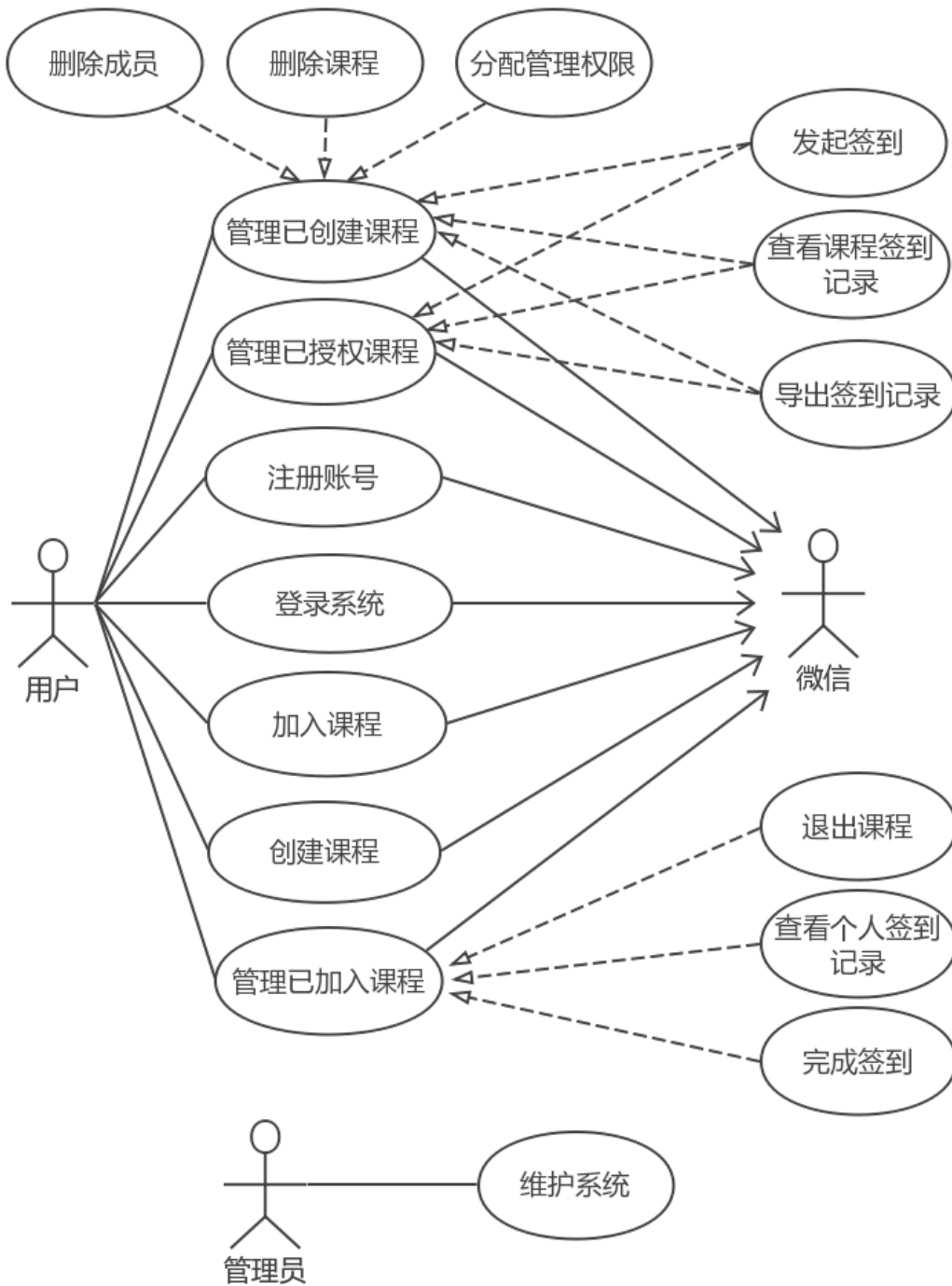
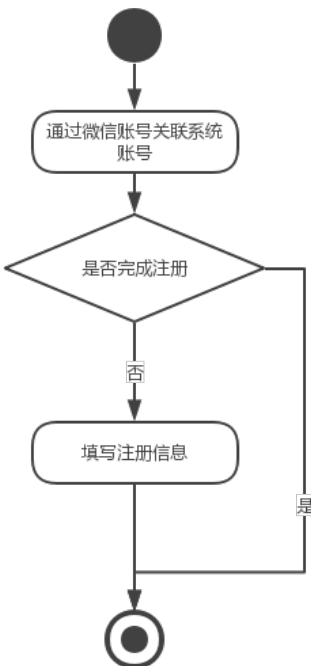


图1-1用例图

2.2.2、活动图

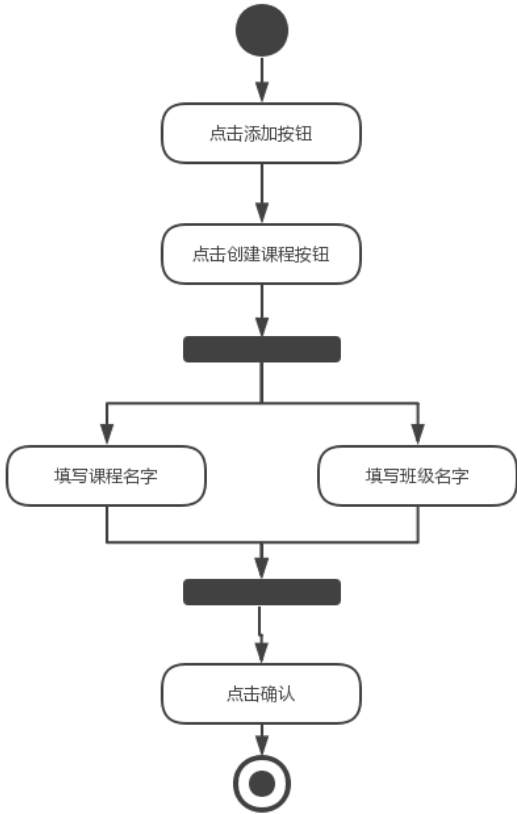
2.2.2.1、注册账号

用例名称	注册账号
简要说明	本系统应用于微信小程序平台，系统账号与微信账号相关联，进入该系统默认使用关联微信账号登陆。在首次登陆时，用户需注册账户的基本信息，包括姓名和学号或工号。
参与者	用户
事件流	1) 用户通过微信和该 APP 系统账号进行关联 2) 判读是否有注册 3) 如果没有注册那么进入注册界面，填写学/工号和姓名 4) 已注册或完成注册用户直接通过账号关联完成登录
特殊需求	无
前置条件	用户使用微信
后置条件	用户可使用系统其余功能
活动图	 <pre>graph TD; Start(()) --> Associate[通过微信账号关系统账号]; Associate --> Decision{是否完成注册}; Decision -- 否 --> Fill[填写注册信息]; Fill --> End((())); Decision -- 是 --> End;</pre> <p>图 1-2 注册账号活动图</p>

2.2.2.2、登录系统

用例名称	登录系统
简要说明	用户打开微信小程序登录系统
参与者	用户
事件流	1) 用户打开微信扫码签到小程序 2) 系统判断用户是否已注册 3) 登录成功
特殊需求	无
前置条件	当前用户已经打开微信
后置条件	用户登录成功后，与后台数据库关联
活动图	<div><pre>graph TD; Start(()) --> Open[打开微信小程序]; Open --> Decision{是否已注册}; Decision -- 是 --> Update[关联数据库更新数据]; Decision -- 否 --> Register[注册账号]; Update --> End((())); Register --> End;</pre></div> <p>图 1-3 登录系统活动图</p>

2.2.2.3、创建课程

用例名称	创建课程
简要说明	用户创建课程需提供课程名字和相关的班级名字。课程创建者获得该课程的管理权限以及对加入课程用户的管理权限。
参与者	用户
事件流	1) 学生登录后可以选择通过扫描二维码或搜索课程 ID 来选择课程 2) 二维码由课程管理者提供，而 ID 则要判断是否合法而且存在该课程 3) 如果合法系统得到并显示该课程内容 4) 学生确定加入课程
特殊需求	无
前置条件	用户完成登陆及注册
后置条件	用户拥有了该课程的创建管理权限
活动图	 <pre>graph TD; Start(()) --> A[点击添加按钮]; A --> B[点击创建课程按钮]; B --> Fork1[]; Fork1 --> C[填写课程名字]; Fork1 --> D[填写班级名字]; C --> Join1[]; D --> Join1; Join1 --> E[点击确认]; E --> End((()));</pre> <p>图 1-4 创建课程活动图</p>

2.2.2.4、加入课程

用例名称	加入课程
简要说明	用户作为学生可以通过课程的二维码或者查询课程 ID，实现加入到该门课程当中。
参与者	用户
事件流	1) 用户登录后可以选择通过扫描二维码或搜索课程 ID 来选择课程 2) 二维码由课程管理者提供，而 ID 则要判断是否合法而且存在该课程 3) 如果合法系统得到并显示该课程内容 4) 用户确定加入课程
特殊需求	无
前置条件	当前用户已经在后台管理同成功登陆
后置条件	用户加入课程后，系统会有确认反馈，而且后台数据库会有所关联
活动图	<pre>graph TD Start(()) --> JoinCourseWay{加入课程方式} JoinCourseWay -- "通过二维码" --> ScanQR[扫描二维码] JoinCourseWay -- "通过ID查询" --> SearchID[搜索课程ID] ScanQR --> WhetherAdd{是否加入课程} SearchID --> IDLegal{ID是否合法} IDLegal -- 是 --> WhetherAdd IDLegal -- 否 --> End(()) WhetherAdd -- 是 --> AddSuccess[加入课程成功] WhetherAdd -- 否 --> End AddSuccess --> End</pre>

图 1-5 加入课程活动图

2.2.2.5、发起签到

用例名称	发起签到
简要说明	课程创建者和课程管理者允许在任何时间发起签到，同时允许设置签到时限和附加验证方式。
参与者	用户
事件流	<ol style="list-style-type: none">1) 用户进入课程管理界面2) 用户选择发起签到3) 用户选择是否修改签到时限，如果选择不修改则采用上一次设置，如果选择修改则用户输入修改后时限。4) 用户选择是否附加 PIN 码验证，如果选择不采用则进入下一事件流，如果选择采用则显示 PIN 码。5) 用户选择是否附加地理验证，如果选择不采用则进入下一事件流，如果选择采用则重新获取用户当前地理位置。6) 用户提交签到设置7) 显示发起签到成功界面
特殊需求	无
前置条件	用户完成登陆、用户为课程创建者或获得课程管理权限。
后置条件	签到时限到后系统会关闭本次签到入口。

活动图

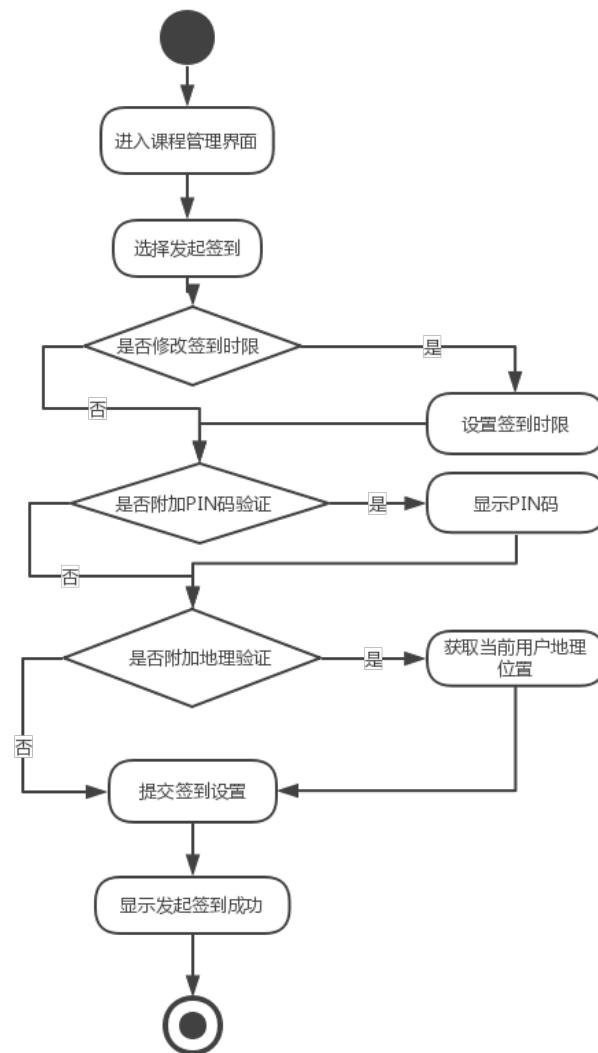


图 1-6 发起签到活动图

2.2.2.6、导出签到记录

用例名称	导出签到记录
简要说明	课程创建者和课程管理者允许将签到记录导出到本地。
参与者	用户
事件流	1) 用户进入课程管理界面 2) 用户选择导出签到记录 3) 用户选择导出文件格式 4) 显示导出文件生成成功和下载文件的 url
特殊需求	无
前置条件	用户完成登陆、用户为课程创建者或获得课程管理权限。
后置条件	无
活动图	<div><pre>graph TD; Start(()) --> A[进入课程管理界面]; A --> B[选择导出签到记录]; B --> C[用户输入邮箱地址]; C --> D[生成导出文件并发送 至用户邮箱]; D --> End((()));</pre><p>The diagram is a UML Activity Diagram titled '发起签到活动图' (Initiating Sign-in Activity Diagram). It starts with a solid black circle (start node) at the top. An arrow points down to a rounded rectangle labeled '进入课程管理界面' (Enter Course Management Interface). Another arrow points down to a second rounded rectangle labeled '选择导出签到记录' (Select Export Sign-in Record). A third arrow points down to a third rounded rectangle labeled '用户输入邮箱地址' (User enters email address). A fourth arrow points down to a fourth rounded rectangle labeled '生成导出文件并发送至用户邮箱' (Generate export file and send to user email). Finally, an arrow points down to a bullseye symbol (end node).</p></div> <p>图 1-7 发起签到活动图</p>

2.2.2.7、查看签到记录

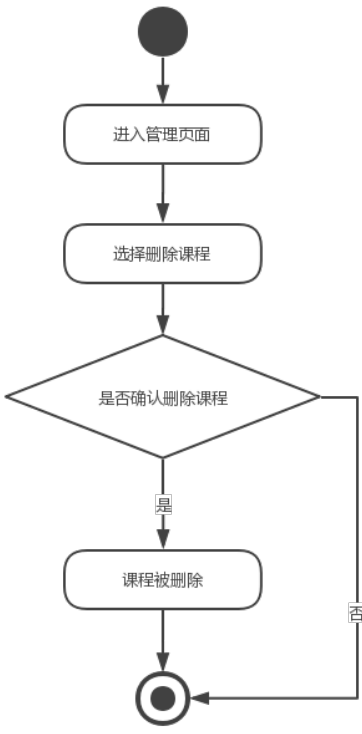
用例名称	查看签到记录
简要说明	用户若以管理员的身份进行管理课程，可以选择查看签到记录，点击查看之后显示出所有加入该课程的用户以及他们具体每一次的签到记录
参与者	用户
事件流	1) 用户点击进入课程 2) 系统判断是否为管理员 3) 如果是管理员则提供查看签到记录选项 4) 管理员选择查看签到记录后显示所有该课程用户的所有签到记录
特殊需求	无
前置条件	用户使用微信
后置条件	用户可使用系统其余功能
活动图	<pre>graph TD; Start(()) --> ClickEnter[点击进入课程]; ClickEnter --> IsAdmin{是否为管理员}; IsAdmin -- 是 --> ClickView[点击查看签到记录]; ClickView --> DisplayResult[显示签到结果]; DisplayResult --> End((())); IsAdmin -- 否 --> End;</pre> <p>图 1-8 查看签到记录活动图</p>

2.2.2.8、删除成员

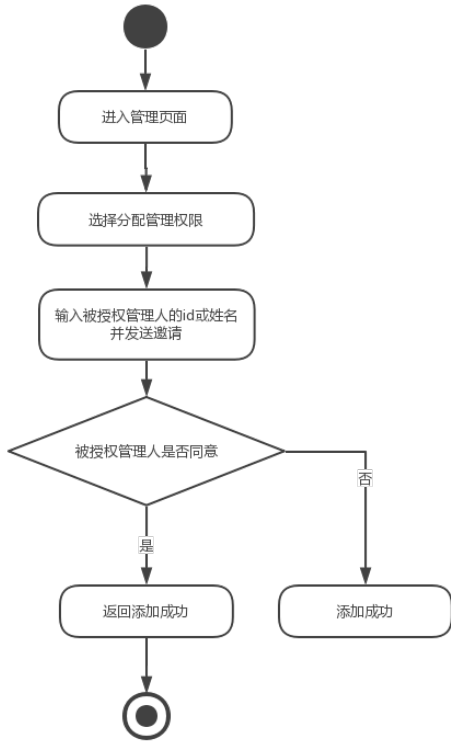
用例名称	删除成员
简要说明	用户若以管理员的身份进行管理课程，可以选择删除成员，选定该课程的某位成员进行删除操作，后台数据库之间的关联也会删掉
参与者	用户
事件流	1) 用户点击进入课程 2) 系统判断是否为管理员 3) 如果是管理员则提供删除成员选项 4) 管理员选择某位成员进行删除 5) 后台数据库进行对应的更改
特殊需求	无
前置条件	用户使用微信
后置条件	用户可使用系统其余功能
活动图	<pre>graph TD; Start(()) --> A[点击进入课程]; A --> B{是否为管理员}; B -- 是 --> C[点击进行成员删除]; B -- 否 --> D(()); C --> D; D --- End((()));</pre>

图 1-9 删除成员活动图

2.2.2.9、删除课程

用例名称	删除课程
简要说明	用户在成为特定课程的创建者或管理者后即具备了对该课程的删除权限，用户可通过点击管理页面的删除课程并点击确认删除按钮后删除课程。
参与者	用户
事件流	1) 用户通过点击首页课程列表中自己所创建或管理的课程进入课程管理页面 2) 用户点击列表最下端的删除课程按钮，系统弹出是否确认删除该课程的选项 3) 用户点击确认，课程被删除。
特殊需求	无
前置条件	用户已完成注册并拥有对课程的管理权，即用户是所需删除课程的创建人或被授权管理人，课程的普通参与者无法使用该功能。
后置条件	无
活动图	 <pre>graph TD; Start(()) --> Enter[进入管理页面]; Enter --> Select[选择删除课程]; Select --> Confirm{是否确认删除课程}; Confirm -- 是 --> Deleted[课程被删除]; Deleted --> End((())); Confirm -- 否 --> End;</pre> <p>图 1-10 删除课程活动图</p>

2.2.2.10、分配管理权限

用例名称	分配管理权限
简要说明	用户在创建课程后可通过分配管理权限邀请课程管理人。
参与者	用户
事件流	1) 用户通过点击首页课程列表中自己所创建的课程进入课程管理页面 2) 用户点击管理列表中分配管理权限按钮 3) 用户输入某用户的 id 或姓名，不可输入自己的 id 或姓名 4) 等待该被邀请用户的确认信息 5) 被邀请用户确认后即邀请成功
特殊需求	无
前置条件	用户已完成注册并是该课程的创建者。
后置条件	被邀请用户确认同意管理
活动图	 <pre>graph TD; Start(()) --> Enter[进入管理页面]; Enter --> Select[选择分配管理权限]; Select --> Input[输入被授权管理人的id或姓名 并发送邀请]; Input --> Decision{被授权管理人是否同意}; Decision -- 否 --> AddSuccess[添加成功]; Decision -- 是 --> ReturnAddSuccess[返回添加成功]; ReturnAddSuccess --> End((()));</pre> <p>图 1-11 分配管理权限活动图</p>

2.2.2.11、完成签到

用例名称	完成签到
简要说明	在课程管理者发起签到后，用户在签到任务指定时间进入签到系统，提供完成签到所需要的验证从而完成签到，验证包括地点认证和 PIN 码认证。地点认证需要用户提供地理位置，PIN 码认证需要用户填写课堂上给出的 PIN 码信息。签到结果将记录到数据统计中。
参与者	用户
事件流	<ol style="list-style-type: none">1) 用户收到课程签到信息2) 用户点击信息进入签到页面3) 用户点击签到，后台通过发起签到者的需求选择对签到者的 pin 码和地点进行选择性的认证4) 如果无误用户进入签到成功界面，否则进入签到失败界面
特殊需求	无
前置条件	用户完成登陆、用户加入了课程且该课程管理者发起了课程签到信息。
后置条件	签到者会收到系统的反馈，包括成功签到、地点错误、pin 码错误、签到取消等反馈，若签到成功需要更新数据统计，若在指定时间内未完成签到则记录为未签到

活动图

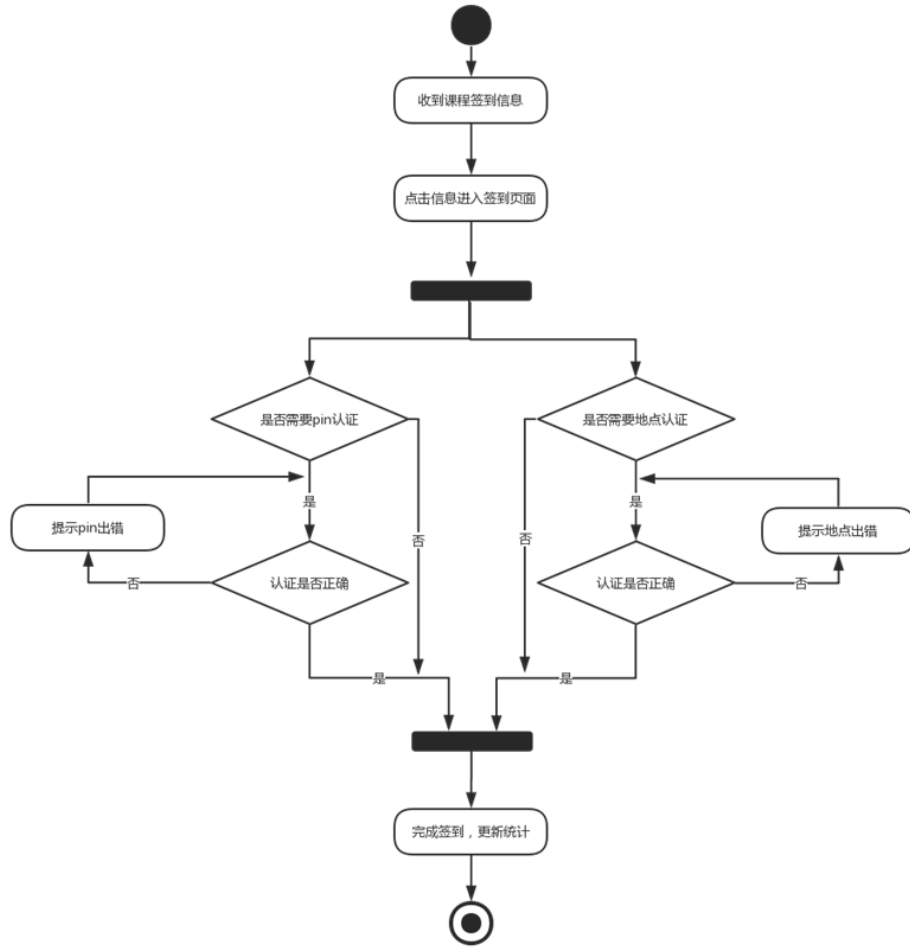


图 1-12 完成签到活动图

2.2.2.12、查看个人签到记录

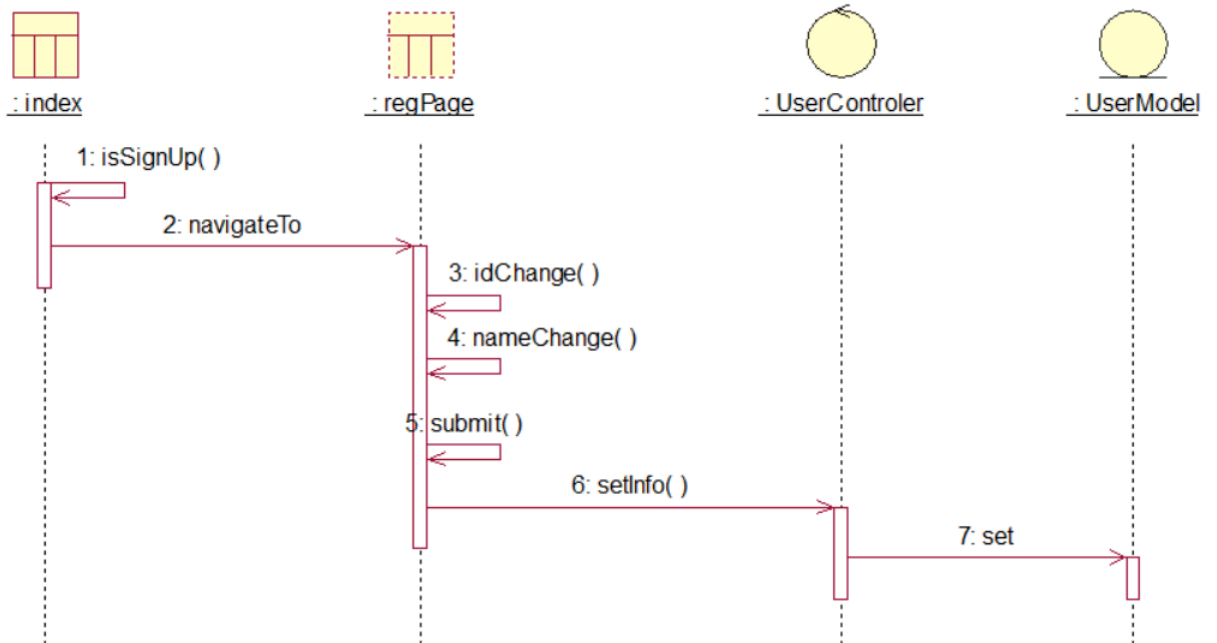
用例名称	查看个人签到记录
简要说明	用户可以查看加入的课程中自己的签到记录或创建的课程的总体签到记录。
参与者	用户
事件流	1) 用户从加入的课程列表中选择课程 2) 系统判断用户是否为课程管理员 3) 显示该课程详细内容 4) 用户查看签到记录
特殊需求	无
前置条件	当前用户已经在后台管理同成功登陆并且已经加入到对应的课程
后置条件	用户结束查看签到记录，后台数据库取消关联
活动图	<pre> graph TD Start(()) --> A[打开已加入课程列表] A --> B[选择需要查看的课程] B --> C{是否为管理员} C -- 是 --> D[显示管理员界面课程信息] C -- 否 --> E[显示学生界面课程信息] D --> F[查看课程签到统计情况] E --> G[查看个人签到统计情况] F --> H(()) G --> H </pre> <p>图 1-13 查看签到情况活动图</p>

2.2.2.13、退出课程

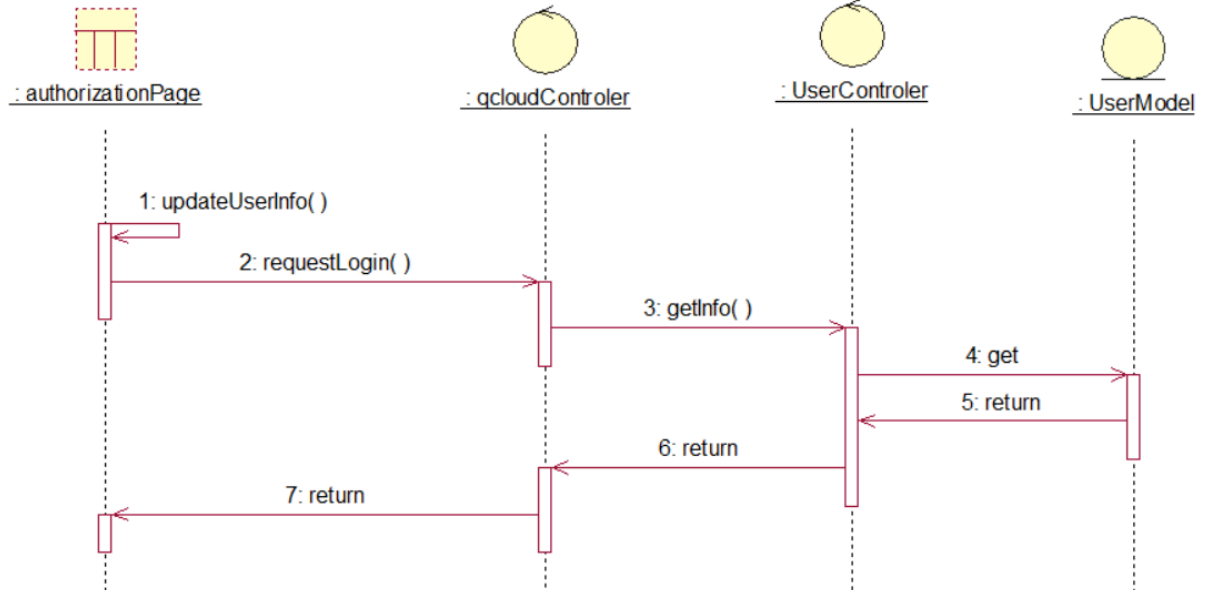
用例名称	退出课程
简要说明	用户可以退出之前已经加入的课程。
参与者	用户
事件流	1) 用户从加入的课程列表中选择课程 2) 系统显示该课程详细内容 3) 用户确认退出课程
特殊需求	无
前置条件	当前用户已经在后台管理同成功登陆并且已经加入到对应的课程
后置条件	用户退出课程后，系统会有确认反馈，而且后台数据库会取消关联
活动图	<pre>graph TD; Start(()) --> Step1[打开已加入课程列表]; Step1 --> Step2[选择欲退出课程]; Step2 --> Step3[退出课程]; Step3 --> End((()));</pre> <p>图 1-14 退出课程活动图</p>

2.2.3、功能模型

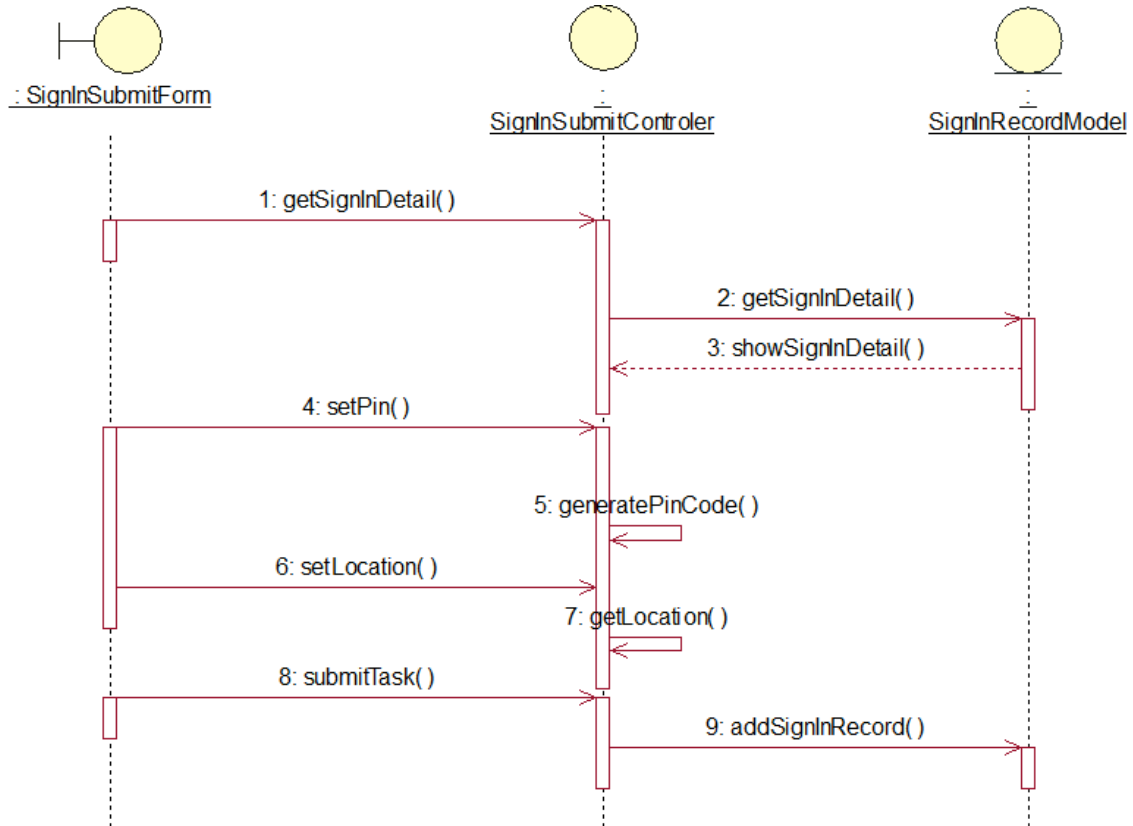
2.2.3.1、用户注册



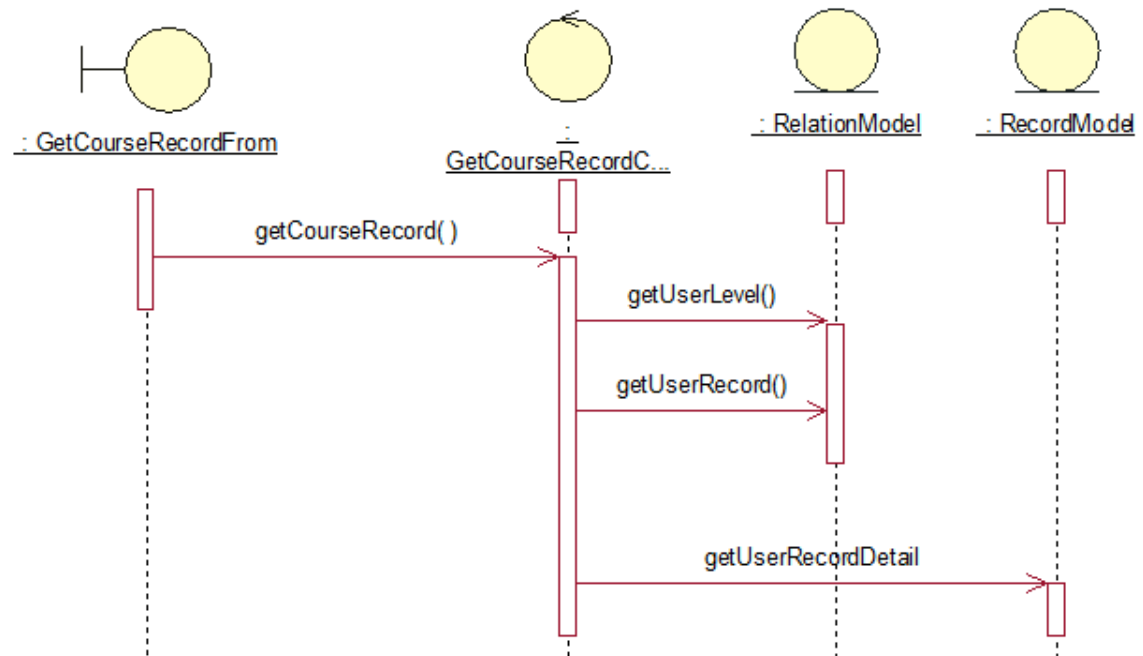
2.2.3.2、登录授权



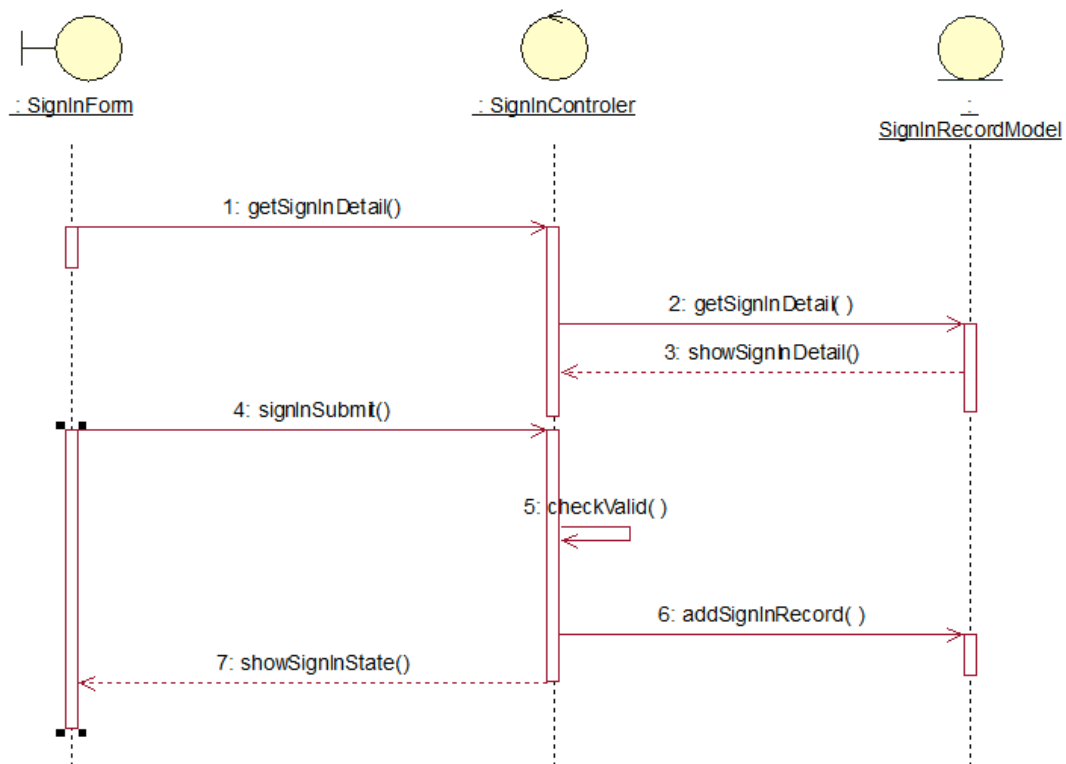
2.2.3.3、发起签到



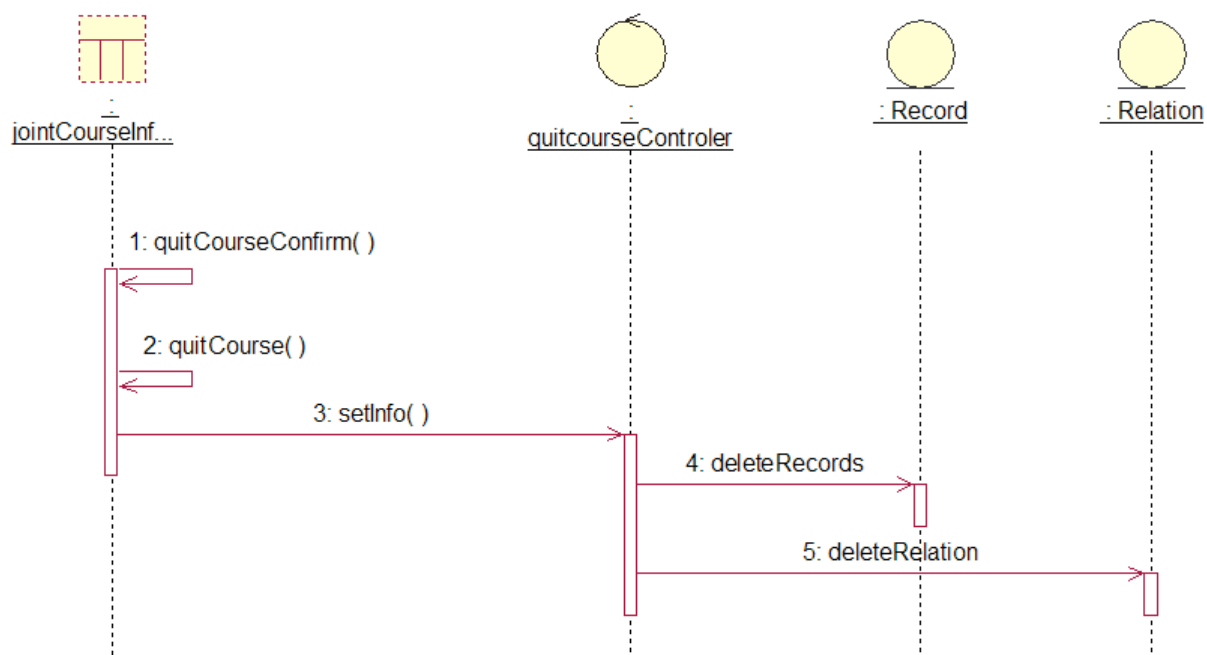
2.2.3.4、查看签到



2.2.3.5、完成签到



2.2.3.6、退出课程



2.3、用户特点

- ✓ 本点餐系统的最终用户为熟悉微信客户端及微信小程序操作的用户，熟悉扫码、下单、移动端支付等操作；
- ✓ 商家管理系统的最终用户为了解电脑操作的商家用户，只需了解如何使用鼠标以及键盘。

2.4、假定与约束

进行本系统开发的约束条件如下：

- ✓ 时间约束：本系统开发周期为3个月，产品经理需要合理规划时间，前后端开发人员需要跟上布置的任务。
- ✓ 实现约束：本系统基于微信小程序Wafer2框架开发，为了方便与微信官方开发工具统一，采用面向对象编程以及前后端均使用Javascript语言。前端直接使用微信小程序组件，后端选型采用Knex与数据库进行连接。

3、需求规定

3.1、对功能的规定

- ✓ 系统内教师和学生以课程为单位进行发起签到和完成签到以及对课程进行相应的管理。本签到系统需要教师和学生通过绑定微信并上传其真实姓名及学/工号成为注册用户，而实名信息由系统管理员维护。
- ✓ 在每门课程开始前，可以由用户创建课程，备注基本课程信息。在成功创建课程后，系统会自动生成唯一识别该课程的课程ID以及二维码。系统允许用户在自己创建的课程内进行删除课程和删除已加入课程的用户的操作，以及把课程管理权限分配加入课程的用户如助教、教务人员等，协同管理课程签到。
- ✓ 用户通过搜索相应的课程ID或扫描二维码加入该课程，如果课程结束或者误加课程，用户可以选择退出课程。
- ✓ 当需要签到时，具有管理权限的用户可以发起签到任务，并自定义签到时限和附加签到验证。若无任何验证，在签到时限内用户进入系统签到界面即可完成该次签到并显示签到成功。如果附加PIN码验证，发起签到的用户则会收到系统随机生成标识该次签到的PIN码，在场的用户通过输入签到发起者公布的PIN码可完成PIN码验证。如果附加地理验证，加入课程用户在进入签到系统后，系统会尝试获取用户的地理位置，如果其地理位置与签到发起者的地理位置在合

理范围内，则会满足地理验证。在签到时限内满足所有验证后，用户进入系统签到界面即可完成该次签到并显示签到成功。

- ✓ 具有管理权限的用户能够随时查看或导出所有已加入课程用户的签到记录，而加入课程的用户可以随时查看自己的历史签到记录。

3.2、对性能的规定

- ✓ 本应用为移动端的web应用，需要有较快的加载速度和页面渲染速度。页面上计算的内容不宜过度，以免造成界面卡顿。在任意既定时刻，系统允许大量用户的并发操作。系统必须能够在2秒钟内完成操作并提供反馈。系统必须能够在2分钟内完成所有事务的 80%。

3.2.1、精度

- ✓ 菜品的价格的精度为小数点后两位，如23.00、100.50等。

3.2.2、时间特性要求

- ✓ 响应时间：
除了初始加载页面的时间在3秒内，签到和加入课程的响应时间都应在1秒内；
数据的转换和传送时间：
- ✓ 数据的转换需在2秒内，传送时间需在1秒内；
- ✓ 同步时间：
用户与其他用户签到情况同步的时间应在用户发起签到后3秒之内；

3.2.3、灵活性

- ✓ 若签到系统由于不可抗力而崩溃，用户可以通过普通签到的方式进行签到；
- ✓ 用户只能在微信小程序上进行签到，老师可以在小程序中查看签到记录也可导出签到记录到邮箱进行查看；

3.3 其他专门要求

3.3.1、可维护性

- ✓ 修改某个模块时，对其他模块的影响和修改代价尽可能小（改动不超过5%）。

3.3.2、可扩展性

- ✓ 允许更多的功能在必要时可以被插入到适当的位置中。

3.3.3、易读性

- ✓ 代码需要有较强的可读性，模块划分清晰，功能明确；
- ✓ 需要有合理的注释和软件开发文档说明。

3.3.4 可用性

- ✓ 方便操作，不需太多学习成本，操作流程合理；
- ✓ 具备容错能力，能拒绝用户的某些不合理的操作

3.3.5、可靠性

- ✓ 系统能够在规定的条件下和规定的时间区间完成规定功能

3.3.6、安全性

- ✓ 系统必须能防止用户在不满足签到要求情况下完成签到。
- ✓ 只有课程创建者能够管理课程。
- ✓ 只有管理员能修改用户注册的实名信息。

3.3.7、可适应性

- ✓ 对于不同品牌和尺寸的设备，如 Apple iPhone X, Mix2 等等，我们需要保证页面一致性。

3.3.8、故障处理

- ✓ 持久性存储中记录错误历史；
- ✓ 正常使用时不应该出错，若运行时遇到不可恢复的系统错误，必须保证数据库完好无损。

4、运行环境

4.1、设备

- ✓ 签到系统需运行在智能触屏手机

4.2、支持软件

- ✓ 签到系统需要运行在微信小程序上