



课堂管理系统

分析与设计

项目组成员：15331004 毕 胜

15331021 陈锦宇

15331232 罗文璐

15331008 曹浩楠

15331079 冯奕健

15331327 谢苑珍

指导老师：衣杨

目录

1.需求分析	3
1.1 问题陈述	3
1.2 用例析取	4
1.3 用例规约	4
● 管理课程用例规约.....	4
● 注册登录用例规约.....	6
● 师生讨论用例规约.....	8
● 发布通知用例归约.....	10
● 选择课程用例规约.....	11
● 课程维护用例规约.....	13
● 人员管理用例规约.....	15
1.4 补充规约	16
1.5 术语表	17
2.架构设计	18
2.1 架构描述	18
2.2 架构图	18
2.3 关键抽象	18
3.用例分析	19
3.1 补充用例规约.....	19
3.2 用例中类的析取.....	19
3.2.1 管理课程用例析取.....	19
3.2.2 注册登录用例析取.....	21
3.2.3 师生讨论用例析取.....	22
3.2.4 发布通知用例析取.....	24
3.2.5 选择课程用例析取.....	26
3.2.6 课程维护用例析取.....	28
3.2.7 人员管理用例析取.....	30
3.3 分析机制	31
3.4 合并分析类.....	32
4.子系统及其接口设计.....	32
4.1 确定设计类.....	32
4.2 定义子系统.....	32
4.3 定义接口	32
4.4 确定可重用子系统	32
5.部件设计	33
5.1 分析并发需求.....	33
5.2 针对加入课程人数问题的解决方案设计.....	33
5.3 生命周期	33
5.4 映射到现实系统.....	33

1.需求分析

1.1 问题陈述

将开发一个新课堂管理系统用于教师在课后管理学生。该系统一共有三个角色，分别是管理员、教师、学生。管理员的功能有课程管理、人员管理；教师的功能有课程管理、发布通知；学生的功能有选择课程，还有一个教师和学生共有的功能：注册登录、查看消息、师生讨论。

教师和学生都可以注册登录该系统，注册需要的资料有学号/工号、密码。学号/工号用于区别学生和教师的角色。教师在学期开始时在该系统中创建所授课程，并设置课堂邀请码，课上教师会提供课堂邀请码保证目标学生的准入。教师在课程结束时可删除课程，该行为会使与该课程相关的信息从数据库中全部清除。教师可发布通知，告知加入该课程的学生作业情况、考核情况、考勤情况、课程时间或内容的调整。教师可针对学生提出的问题进行解答疑问，该解答是面向全体学生的，每个解答都会在其前面附上相应的问题，学生之间也可以进行提问解答，老师也可以在其中进行问题提出。

学生在学期开始时根据获得的课堂邀请码加入相应教师所授的课程。学生在学期结束时可退出课程，该行为会删除学生的已选课程列表中该课程项，但不会删除学生在该课程中的信息。学生可以针对作业或者上课的内容提出疑问，问题是面向全体学生和教师的。学生有每个课程有消息列表，可以查看以往的通知、问题以及相应的解答。

管理员可在系统中管理课程的信息以及管理教师和学生的账号。

1.2 用例析取

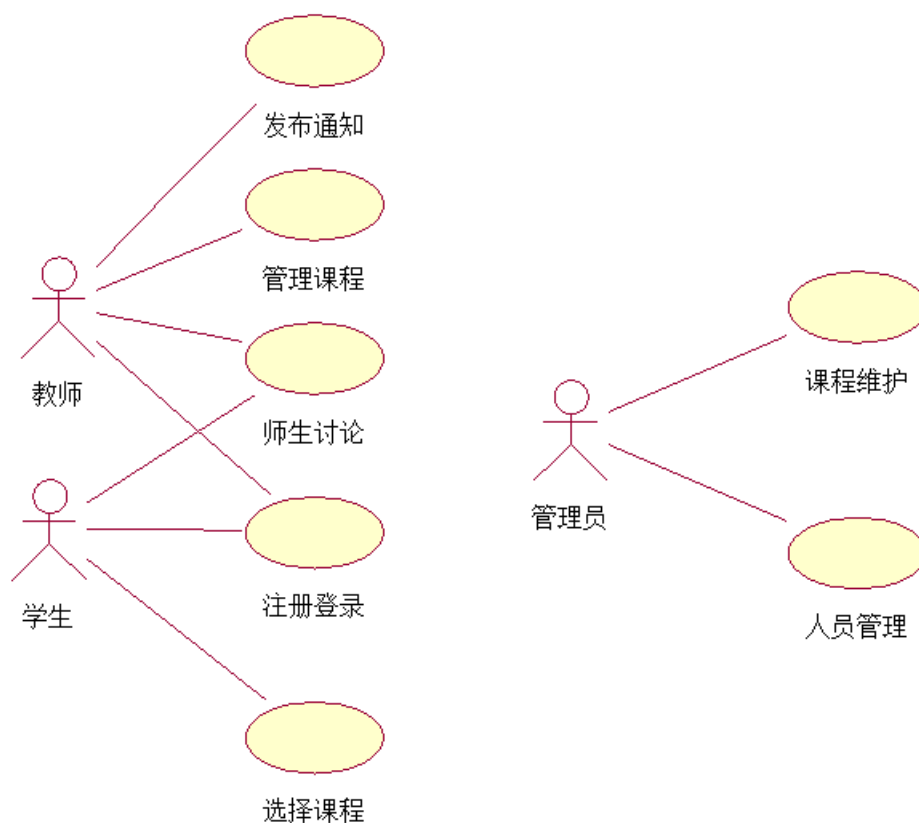


图 1-1 课程管理系统用例图

1.3 用例规约

● 管理课程用例规约

1. 简要说明

本用例允许教师管理本学期的课程。在学期开始时，教师可以创建或者删除所授课程。

2. 事件流

2.1 基本事件流

用例开始于教师管理的课程

1) 系统要求教师指出要执行的操作（创建课程、修改课程、删除课程）

2) 一旦教师提供了所需要的信息，以下的一条子事件流将被执行

如果选择的是“创建课程”，创建课程子事件流将被执行

如果选择的是“删除课程”，删除课程子事件流将被执行

2.1.1 创建课程

- 1)系统显示当前教师已经创建的课程列成表格，并将列表显示给教师
- 2)一旦教师决定了创建课程信息，系统为他创建一个课程
- 3)执行提交课程子事件流

2.1.2 提交课程

对于课程列表中每门未标有“已创建”的课程，系统检验教师满足必要的预备条件，并且教师确定提交后，系统将教师添加的课程显示在课程列表中，同时系统会给教师一个课堂邀请码，这个课程在课程列表中表明“已创建”。系统保存到课程列表。

2.1.3 删除课程

- 1) 系统从根据教师已经创建的课程列表，并将列表显示给教师
- 2) 一旦教师点击了删除课程信息，系统弹出对话框询问是否删除
- 3) 课程信息完全删除，数据库中关于该课程的信息全部清空

2.2 备选事件流

2.2.1 保存课程

在任何时候，教师都可以选择保存课程而不提交它。如果这样，提交课程的步骤将被以下步骤所取代：课程在教师课程列表中的标识不是“已创建”而是“已保存”，系统保存课程表，但不显示在学生课程列表中。

2.2.2 未找到课程

如果在修改课程表子事件流中，系统无法返回该课程，将显示错误消息。教师确认错误，这时本用例重新开始。

2.2.3 回收课程

在任何时候，教师都可以选择回收课程而不删除它。如果这样，提交课程的步骤将被以下步骤所取代：课程从教师课程列表中删除而进入回收站，课程信息不变，教师之后可以点击恢复。

3. 特殊需求

无。

4. 前置条件

本用例开始前教师必须已经登陆进系统，系统能够正常运行。

5. 后置条件

如果用例成功，教师管理的课程信息将会发生改变。否则系统状态不变。

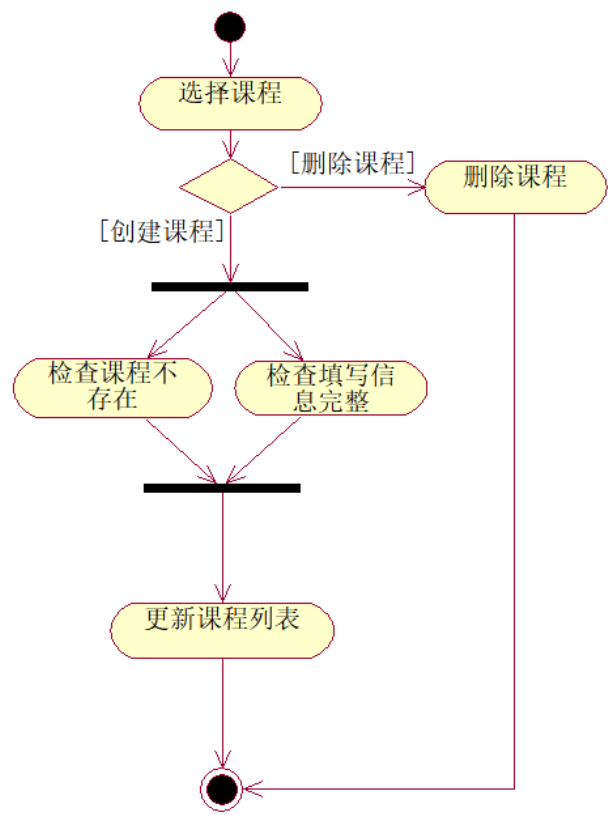


图 1-2 管理课程用例活动图

● 注册登录用例规约

1. 简要说明

本用例允许教师和学生在该课堂管理系统中注册账号与登录账号。

2. 事件流

2.1 基本事件流

用例开始于教师/学生顺利开启系统进入账号登录界面

1) 系统要求教师/学生指出要执行的操作（注册账号、登录账号）

2) 一旦教师/学生提供了所需要的信息，以下的一条子事件流将被执行

如果选择的是“注册账号”，注册账号子事件流将被执行；

如果选择的是“登录账号”，登录账号子事件流将被执行

2.1.1 注册账号

1) 系统提供两个选项让用户选择注册教师或者学生账号

2)用户选择后，进入填写注册资料的页面，资料包括：学号/工号、密码。

3)资料填写无误，系统为该用户创建一个教师或者学生账号

2.1.2 登录账号

1) 系统显示出登录界面，需要教师/学生填入所需的资料

2) 教师/学生填写资料：学号/工号、密码。

3) 系统根据学号和工号区别开教师与学生的身份，以此进入不同的个人界面

2.2 备选事件流

2.2.1 注册时学号/工号不合规范

注册时，学号/工号的格式（长度、类型）错误，系统无法进行该账号的注册，将显示错误信息。用户确认错误，重新输入学号/工号

2.2.2 注册时密码不合规范

注册时，输入的密码不符合以下规范：密码长度 8-12 位，必须包括字母和数字。否则系统无法进行该账号的注册，将显示错误信息。用户确认错误，重新输入密码

2.2.3 登录时学号/工号错误

登录时，学号/工号的格式不符合或系统找不到该学号/工号对应的账户，系统无法通过该账户的登录，将显示错误信息。用户确认错误，重新输入学号/工号

2.2.4 登录时密码错误

登录时，密码输入错误，系统无法将输入的学号/工号与密码匹配，此时，系统无法进行该账号的登录，将显示错误信息。用户确认错误，重新输入密码

3. 特殊需求

无。

4. 前置条件

本用例开始前用户必须已经成功启动系统。

5. 后置条件

如果用例成功，以下两个结果将会产生：

若选择的是“注册账号”，系统将为该用户生成一个对应的教师/学生账户，并跳转到对应的个人界面；

若选择的是“登录账号”，系统跳转到对应的教师/学生个人界面

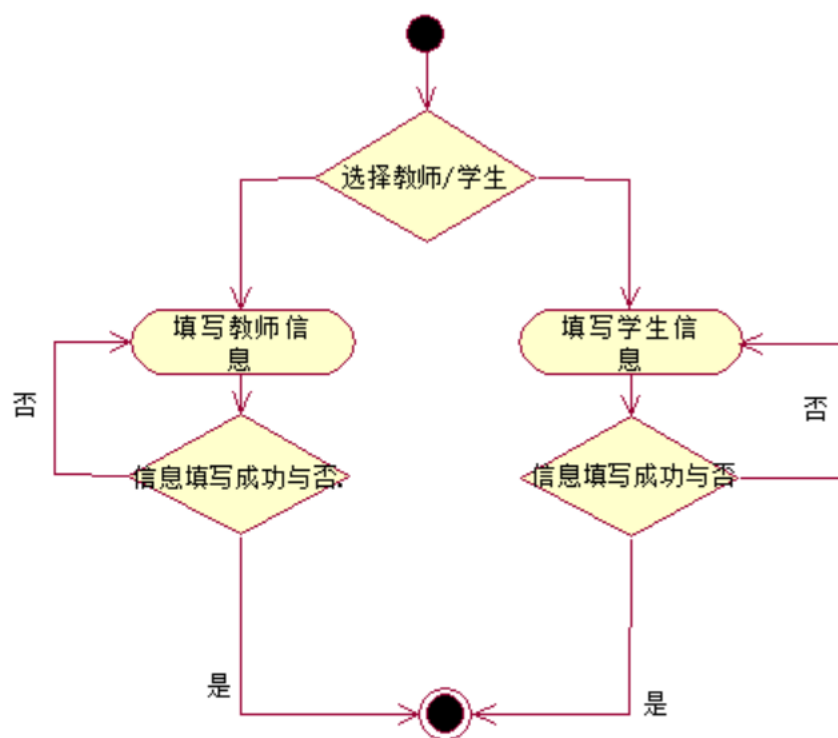


图 1-3 注册登录用例活动图

● 师生讨论用例规约

1. 简要说明

本用例允许学生老师提出疑问，学生和教师都可以进行解答。在该系统中，学生可以提出关于该课程某项通知或者对于课堂内容的疑问，教师可以对同学们的一些情况进行必要的提问，教师可以查看学生针对该课程内容的疑问列表，学生可以对老师或同学之间的问题进行回答。

2. 事件流

2.1 基本事件流

用例开始于教师/学生在该课程中选择提出疑问

- 1) 系统要求教师/学生指出要执行的操作（提出疑问）
- 2) 一旦教师/学生提供了所需要的信息，以下的一条子事件流将被执行

如果选择的是“提出疑问”，提出疑问子事件流将被执行

2.1.1 提出疑问

- (1) 教师/学生对某个通知提出疑问
- (2) 教师/学生在疑问框内以文字或图片进行编辑

(3) 提出疑问的结果会告知该课程的所有学生和教师，出现在学生的信息列表和教师的疑问列表中。疑问会在其上附加本次通知的内容。

3) 系统要求教师/学生指出要执行的操作（解答疑问）

4) 一旦教师/学生提供了所需要的信息，以下的一条子事件流将被执行

如果选择的是“解答疑问”，解答疑问子事件流将被执行

2.1.2 解答疑问

系统将根据教师/学生提出的疑问合成列表，并将列表显示给教师/学生

教师/学生点击疑问，并用文字或图片的形式回复该问题

解答疑问的结果会通知该课程的所有学生以及教师，出现在教师/学生的信息列表中

2.2 备选事件流

2.2.1 针对课堂内容提出疑问

若教师/学生不是针对某次通知，而是针对课堂内容或学习方法或掌握程度提出疑问，需要教师/学生填写的疑问框中不会在前面附上通知的内容，而只是显示提出的疑问

2.2.2 疑问列表为空

若教师/学生在疑问列表找不到疑问，也即教师/学生没有针对该课程的问题提出疑问。则疑问列表为空，教师/学生也无法进行解答。

3. 特殊需求

无。

4. 前置条件

本用例开始前教师/学生必须已经登录进系统，并且已经加入了该课程。

5. 后置条件

如果用例成功，教师/学生查看课程的信息列表出现教师/学生提出的疑问；全体学生/教师接收到这次疑问内容，其出现在教师/学生的信息列表中。教师/学生查看的疑问列表中，该消息自动归为“已阅读”类型，否则系统状态不变。

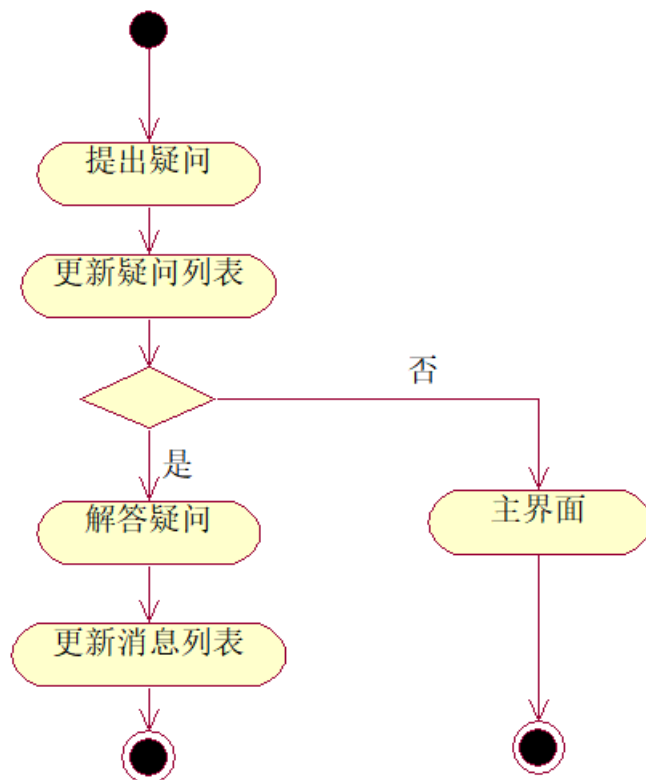


图 1-4 师生讨论用例活动图

● 发布通知用例归约

1. 简要说明

本用例允许教师根据自身的教学需求，发布对应课程所有学生必会收到的课程信息，包括课程通知。

2. 事件流

2.1 基本事件流

用例开始于教师对进行自己的课程，查看选项。

1) 系统要求教师给出要执行的操作。

2) 教师给出操作，以下子事件流将被执行。

如果选择的是发布课程信息，发布课程信息子事件流将被执行。

2.1.1 发布课程信息

1) 输入通知标题和内容

2) 发布通知

2.2 备选事件流

无。

3.特殊需求

无。

4.前置条件。

作为教师身份进入课程。

5.后置条件。

服务器接收信息，发送给学生端。

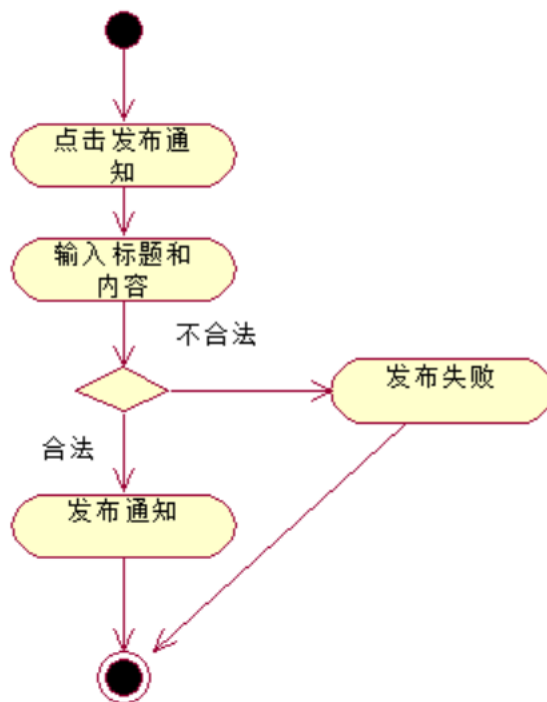


图 1-5 教师发布通知用例活动图

● 选择课程用例规约

1. 简要说明

本用例允许学生选择加入或者退出本学期的课程。在学期开始时，学生可以使用课堂邀请码加入所选课程，在学期结束时，学生可以退出所选课程。

2. 事件流

2.1 基本事件流

用例开始于学生查看所选的课程

1) 系统要求学生指出要执行的操作

2) 一旦学生提供了所需要的信息，以下的一条子事件流将被执行

如果选择的是“加入课程”，加入课程子事件流将被执行

如果选择的是“退出课程”，退出课程子事件流将被执行

2.1.1 加入课程

- 1) 系统将教师创建的课程整理成一个课程列表，并将列表显示给学生
- 2) 一旦学生决定了加入该课程，则点击对应的课程
- 3) 执行课堂准入子事件流

2.1.2 课堂准入

学生在弹出的对话框中输入该课堂的课堂邀请码，成功加入该课堂。

2.1.3 退出课程

- 1) 系统从根据学生已经加入的课程列表，并将列表显示给学生
- 2) 一旦学生点击了退出课程，系统弹出对话框询问是否退出
- 3) 若选择退出，学生的已加入课程列表中删除该课程的选项

2.2 备选事件流

2.2.1 课堂邀请码错误

若学生输入了错误的课堂邀请码，系统则不通过该学生对该课堂加入的申请，并提示错误，需要学生重新选择加入课堂

2.2.2 退出后重新加入

若学生退出课程后想要获得该课程的信息，希望能重新加入，只要输入正确的课堂邀请码，系统会匹配账号，若账号不变，匹配成功，则能够加入该课程并且获得原来的信息。

3. 特殊需求

无。

4. 前置条件

本用例开始前学生必须已经登录进系统。

5. 后置条件

如果用例成功，以下两个结果将会产生：

若选择的是“加入课程”，该课程则加入该学生的信息；

若选择的是“退出课程”，学生的课程列表中该课程的选项被删除，课程中该学生的状态显示“已退出”，但数据库中关于该学生在该课程的信息不会清空。

否则系统状态不变。

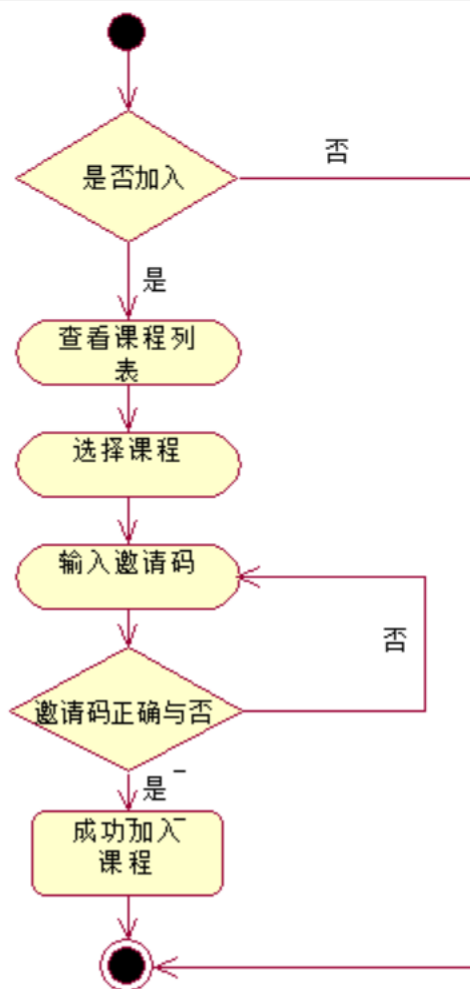


图 1-6 选择课程用例活动图

● 课程维护用例规约

1. 简要说明

本用例允许管理员维护每个课程产生的所有信息

2. 事件流

2.1 基本事件流

用例开始于管理员要查看每个课程的信息

1) 系统要求管理员指出要执行的操作（课程信息管理）

2) 一旦管理员提供了所需要的信息，以下的一条子事件流将被执行

如果选择的是“课程信息管理”，课程信息管理子事件流将被执行

2.1.1 课程信息管理

- 1) 系统显示课程列表，该列表包含所有的课程
- 2) 管理员可以点击任意一个课程查看课程中出现的所有信息
- 3) 管理员可将课程的信息导出或者增添、删除

2.2 备选事件流

2.2.1 修改错误信息

若学生提出的疑问或教师发布的通知、解答出现错误，情形严重者可委托管理员将其修改或者删除

2.2.2 管理员通知

管理员若进行系统维护需要暂停系统的使用，可对所有教师和学生发出通知

3. 特殊需求

无。

4. 前置条件

本用例开始前管理员必须能进入该系统。

5. 后置条件

如果用例成功，所有课程的信息按照管理员的操作进行增添、修改、删除、导出。否则系统状态不变。

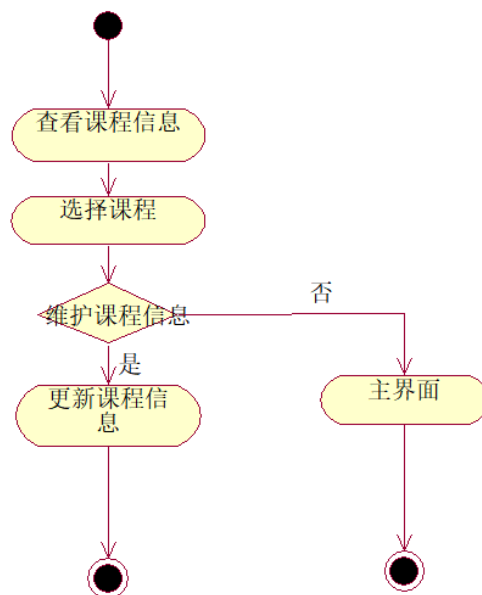


图 1-7 课程维护用例活动图

● 人员管理用例规约

1. 简要说明

本用例允许管理员管理系统中注册的所有用户，包括所有教师和学生

2. 事件流

2.1 基本事件流

用例开始于管理员要查看每个用户的信息

1) 系统要求管理员指出要执行的操作（人员管理）

2) 一旦管理员提供了所需要的信息，以下的一条子事件流将被执行

如果选择的是“人员管理”，人员管理子事件流将被执行

2.1.1 人员管理

1) 系统显示所有用户，包括所有的教师和学生，显示为两个列表

2) 管理员可以点击任意一个用户查看其所有信息和资料

3) 管理员可将人员的信息导出或者增添、删除

2.2 备选事件流

2.2.1 修改错误信息

若学生登记的个人资料出现错误，情形严重者可委托管理员将其修改或者删除

3. 特殊需求

无。

4. 前置条件

本用例开始前管理员必须能进入该系统。

5. 后置条件

如果用例成功，所有人员的信息按照管理员的操作进行增添、修改、删除、导出。

否则系统状态不变。

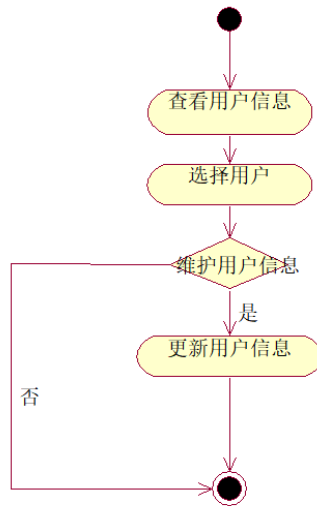


图 1-8 人员管理用例活动图

1.4 补充规约

1. 目标

本文档的目的是定义课堂管理系统的需求。本补充规约列出了不便于在用例模型的用例中获取的系统需求。补充规约和用例模型一起记录关于系统的一整套需求。

2. 范围

- 1) 本补充规约适用于课堂管理系统,将要由学习面向对象软件分析与设计的学生开发。
- 2) 本规约除定义了在许多用例中所共有的功能性需求以外,还定义了系统的非功能性需求,例如:可靠性、可用性、性能和可支持性等。(功能性需求在用例规约中定义。)

3. 参考

无

4. 功能

- 1) 多个用户必须能同时执行操作。
- 2) 如果某教师/学生输入信息错误,系统必须提醒该教授/学生
- 3) 发布的信息必须告知所有的教师/学生

5. 可行性

手机用户界面应与 Android 7.0 兼容。

6. 可靠性

选课系统在每周七天,每天二十四小时内都应是可以使用的。宕机的时间应少于 10%。

7. 性能

- 1) 在任意既定时刻,系统最多可支持 2000 名用户同时使用中央数据库,并在任意时刻最多可支持 500 名用户同时使用本地服务器。
- 2) 系统将能在十秒钟内提供对遗留课程目录数据库的访问。
- 3) 注意:基于风险的原型发现遗留课程目录数据库在没有利用中层处理能力的前提下,无法满足性能上的需求。
- 4) 系统必须能够在 2 分钟内完成所有事务的 80%。

8. 可支持性

无

9. 安全性

- 1) 系统必须能防止学生修改他人的资料以及教师修改其他教师所创建的课程信息。
- 2) 只有教师能发布通知。
- 3) 只有管理员能修改学生的信息。

10. 设计约束

系统必须提供基于 Android 7.0 的接口。

1.5 术语表

名词术语	定义	英文名
教师	教授他人课程的主体	Teacher
学生	接受参与课程的主体	Student
管理员	管理以及维护系统的主体	Manager
工号	教师的唯一标识,可用于注册账号	Job ID
学号	学生的唯一标识,可用于注册账号	Student ID
信息	课程中出现的所有文字和图片	Information
资料	描述个人的文字材料	Data
疑问	学生提出的问题	Question
解答	教师针对问题提出的答案	Answer
通知	教师向该课程内全体学生发布的问题和图片信息	Notice

2. 架构设计

2.1 架构描述

本系统采用 MVC（Model-View-Controller）模式进行架构设计。该系统分为模型层、视图层和控制器层。MVC 架构设计将业务逻辑、数据和界面显示分离开来，使得开发人员在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。

视图层主要是课程管理系统图形界面，为用户提供交互界面；控制器负责转发请求，对请求进行处理，它控制着应用程序的流程，处理时间并进行响应；模型封装了与应用程序的业务逻辑相关的数据如课程信息、用户信息等。

2.2 架构图

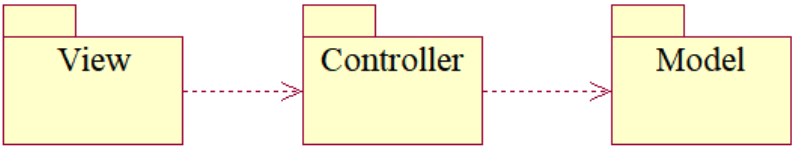
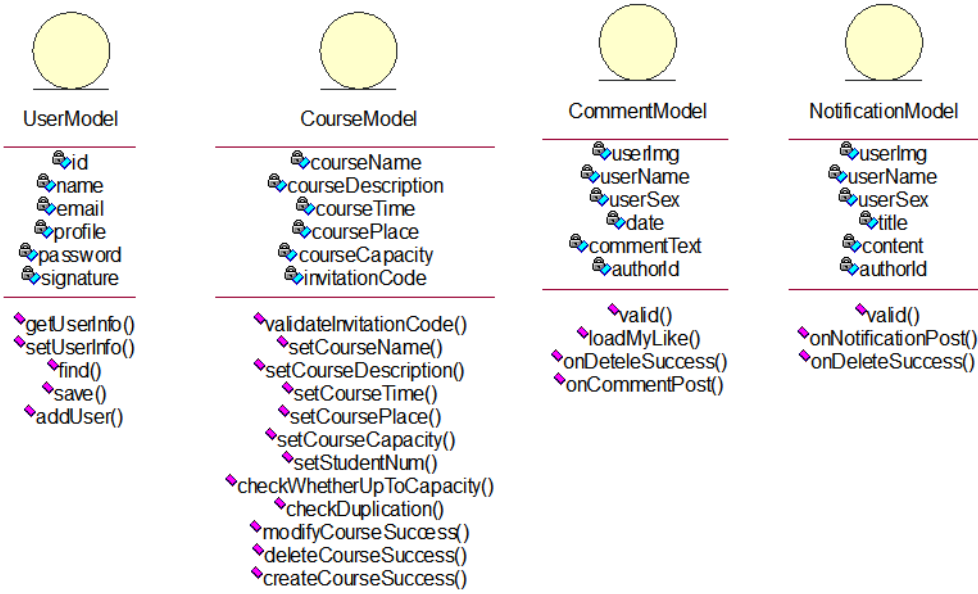


图 2-1 课堂管理系统架构图

2.3 关键抽象

经过我们分析，课程管理系统的关键抽象如下图所示：



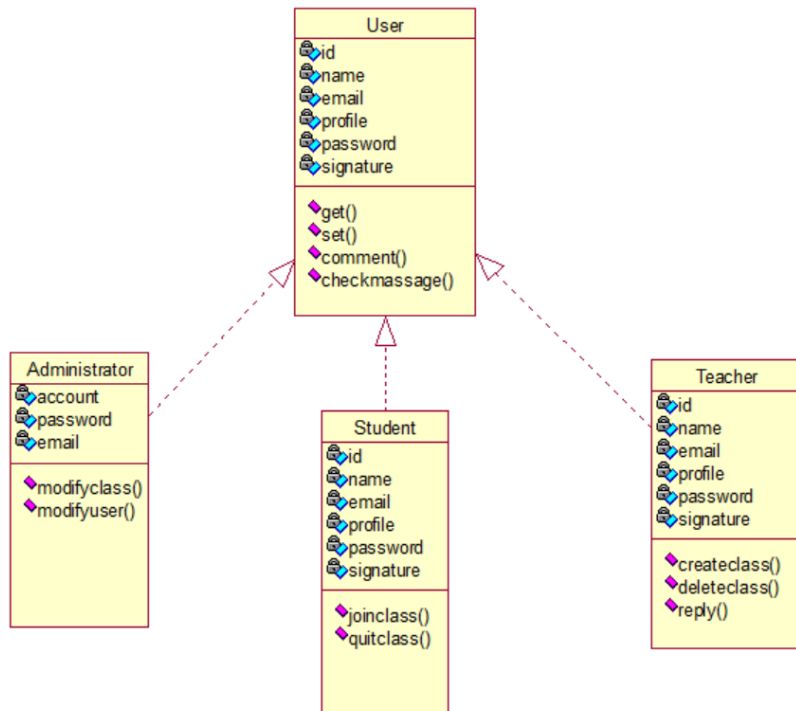


图 2-2 课堂管理系统关键抽象

3.用例分析

3.1 补充用例规约

经检查，本项目组发现用例规约比较完善，暂时无需补充。

3.2 用例中类的析取

3.2.1 管理课程用例析取

根据需求分析中的用例归约，经分析可得到该用例中的三种类。

边界类：courseForm。courseForm 为课程管理页面。

控制类：CourseAction。CourseAction 控制类负责处理管理课程的相关操作，包括查看所有课程，创建新课程，修改课程，删除课程。

实体类：CourseModel。CourseModel 实体类表示课程实体，包含课程名称，课程简介（包含课程老师），上课时间，上课地点，课程容量，已选人数信息。

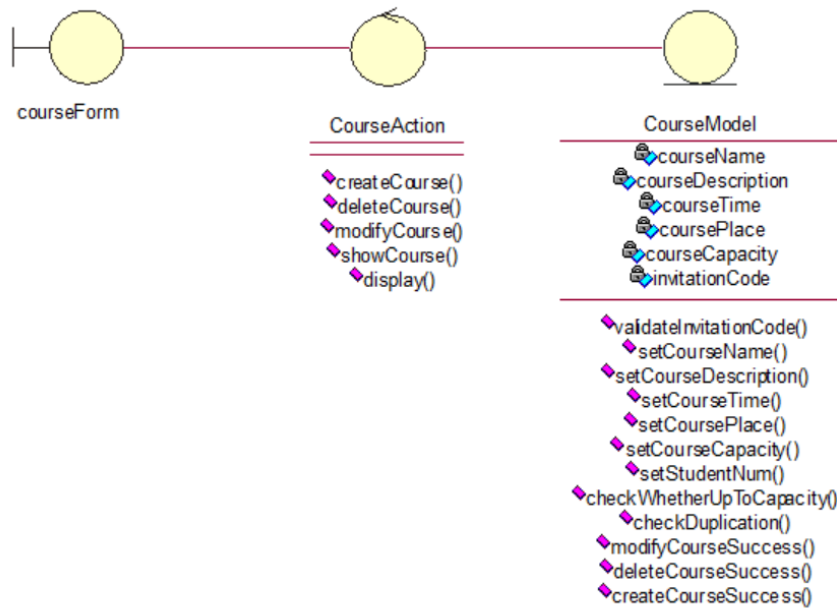


图 3-1 管理课程用例析取图

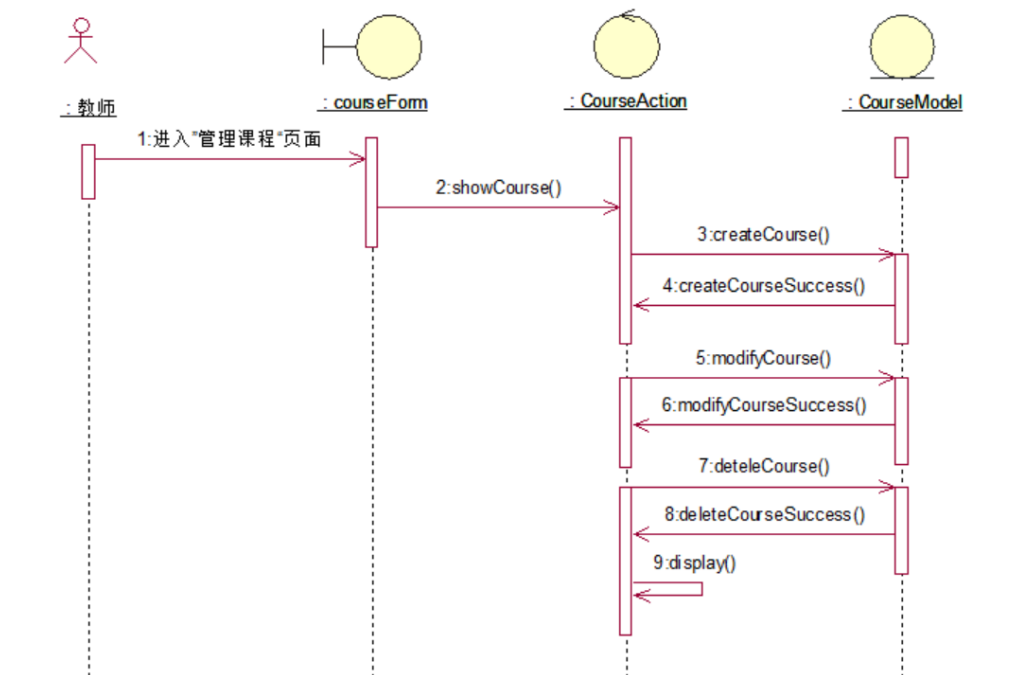


图 3-2 管理课程用例时序图

教师首先点击进入“管理课程”的页面，调用 `showCourse()`来显示目前所有的课程。创建课程调用 `createCourse()`并输入一系列的课程信息。如果创建成功则返回 `createCourseSuccess()`。修改课程调用 `modifyCourse()`，如果修改成功则返回 `modifyCourseSuccess()`，最后如果要删除课程，调用 `deleteCourse()`，如果成功则返回 `deleteCourseSuccess()`，通过 `display()`来显示最新的课程情况。

3.2.2 注册登录用例析取

根据需求分析中的用例归约，经分析可得到该用例中的三种类

边界类：SignUpForm 和 SignInForm。SignUpForm 是注册页面，SignInForm 是登录页面。

控制类：SignAction。SignAction 控制类负责处理注册和登录的相关操作，包括：输入学号/工号，输入密码，再次输入密码，选择学生或老师的身份。

实体类：UserModel。UserModel 实体类表示注册用户实体（被 teacher 和 student 类所继承），包含：学号/工号，密码，邮件，昵称，个性签名，性别。

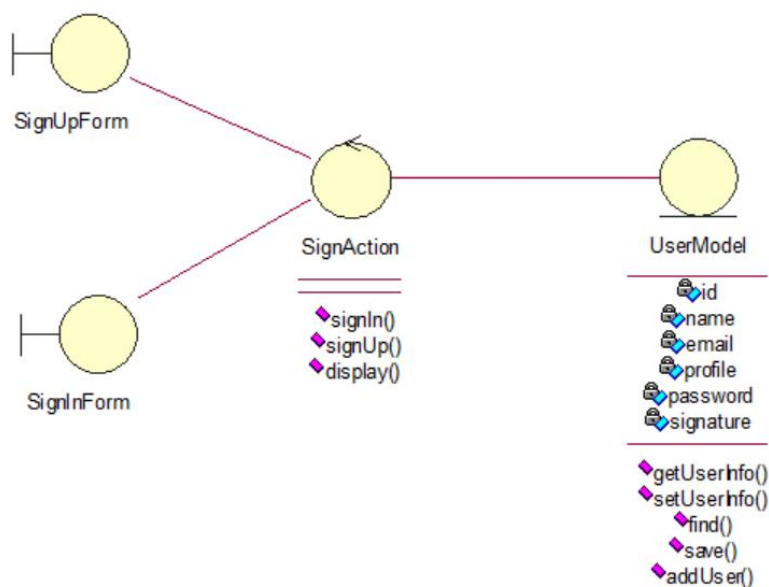


图 3-3 注册登录用例析取图

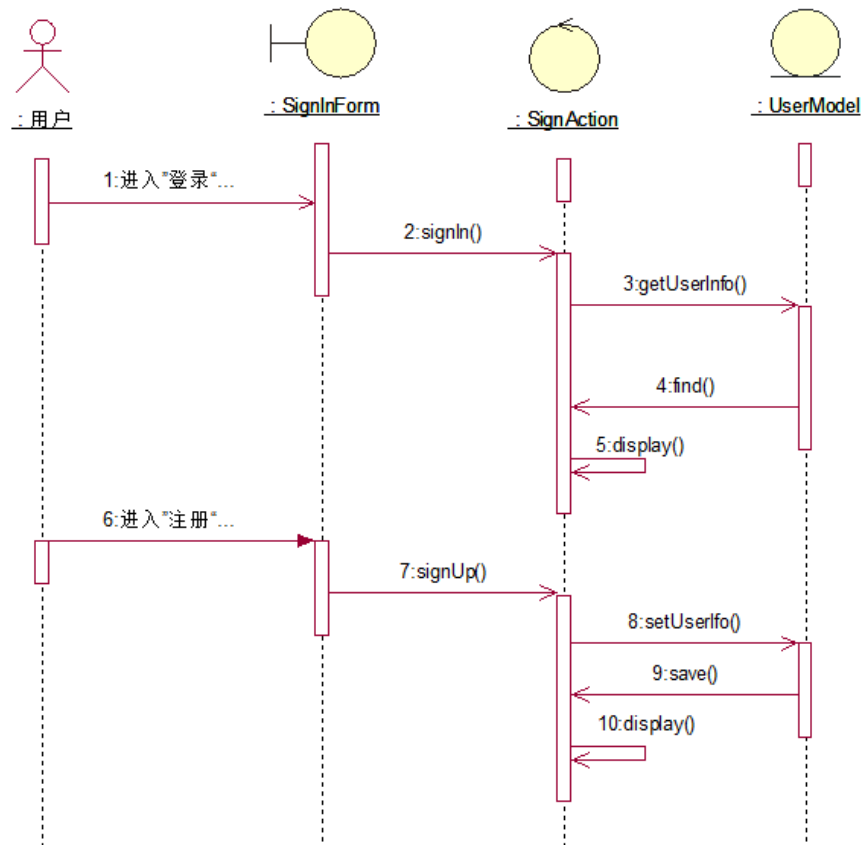


图 3-4 注册课程用例时序图

用户可以选择登录或者注册。首先是进入登录界面。调用 `signIn()` 进行登录。首先调用 `getUserInfo()` 来寻找是否已经有登录过的痕迹。通过 `find()` 返回，输入信息之后，`display()` 刷新登录界面的信息。新用户可以注册，首先是调用 `signUp()` 进行注册。调用 `setUserInfo()` 输入注册的新用户的信息。然后 `save()` 返回保存结果，最后 `display()` 刷新注册界面信息。

3.2.3 师生讨论用例析取

根据需求分析中的用例归约，经分析可得到该用例中的三种类

边界类：`CommentForm`。`CommentForm` 是师生讨论区页面。

控制类：`CommentAction`。`CommentAction` 控制类负责讨论的相关操作，包括：发表帖子，输入帖子内容，查看帖子，评论帖子，收藏帖子

实体类：`CommentModel`。`CommentModel` 实体类表示讨论的师生实体，包括：用户个人信息、用户昵称、用户性别、评论日期、评论内容、作者 ID。

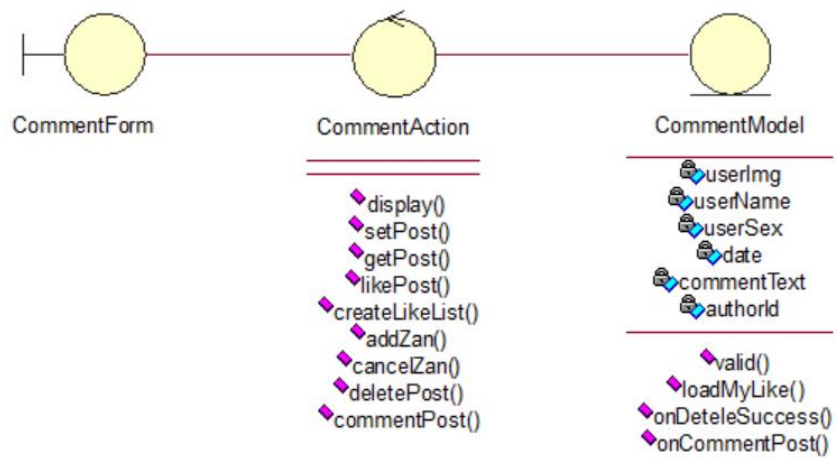


图 3-5 师生讨论用例析取图

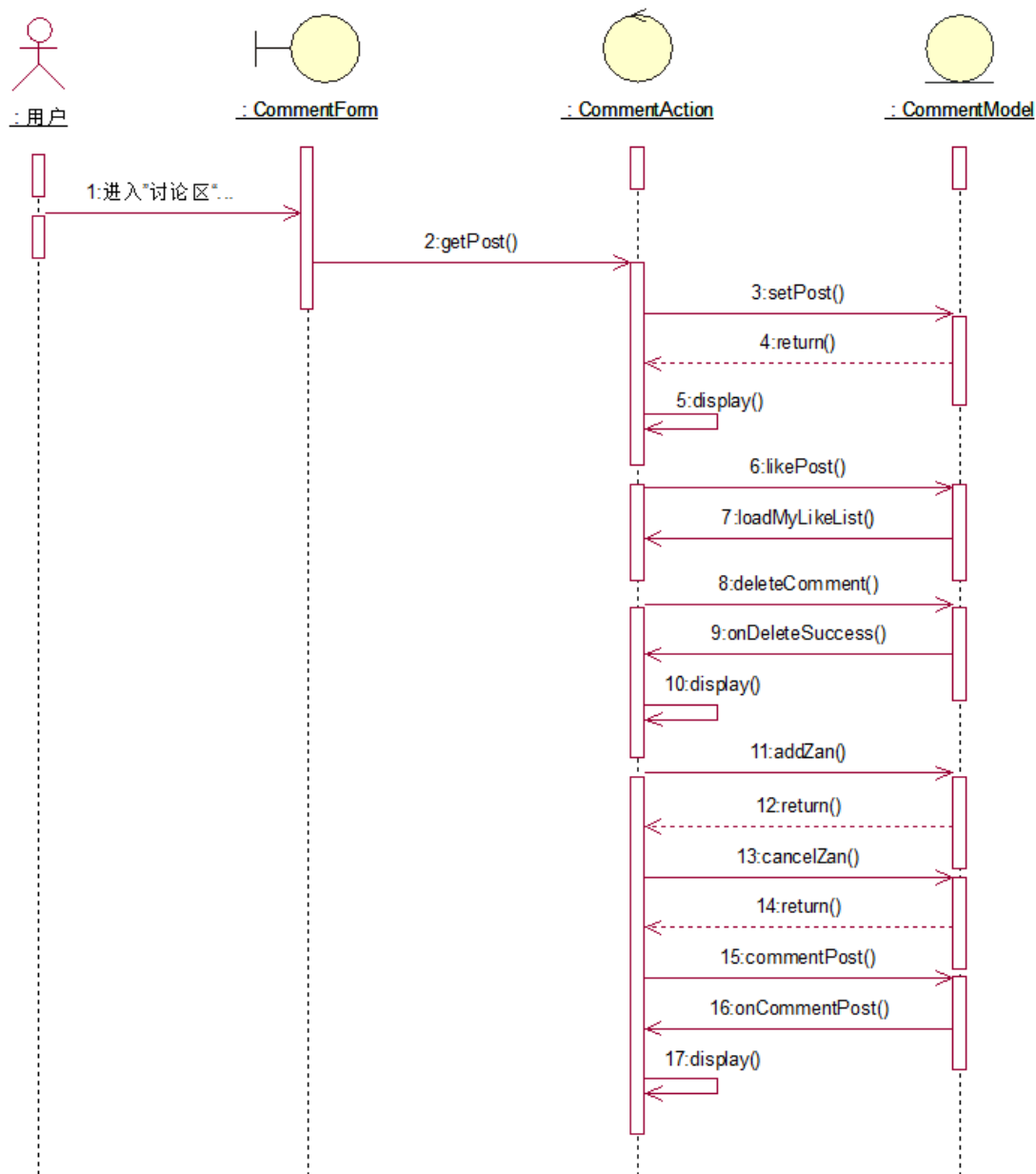


图 3-6 师生讨论用例时序图

用户首先点击进入“讨论区”，调用 `getPost()` 查看讨论区的内容。发表帖子则调用 `setPost()` 编辑帖子的内容。用 `return()` 返回。`display()` 刷新界面。收藏帖子调用 `likePost()` 成功返回 `loadMyLikeList()` 显示收藏列表。删除评论调用 `deleteComment()`，成功后返回 `onDeleteSuccess()`，点赞帖子调用 `addZan()`，成功后 `return()`。评论帖子调用 `commentPost()`，成功后返回 `onCommentPost()`，最后调用 `display()` 刷新评论区页面。

3.2.4 发布通知用例析取

根据需求分析中的用例归约，经分析可得到该用例中的三种类边界类：`NotificationForm`。`Notification` 是通知显示区页面。

控制类：NotificationForm。Notification 控制类负责通知的相关操作：发布通知，输入通知的标题和内容，查看通知，标记已读通知。

实体类：NotificationModel。NotificationModel 实体类表示发布通知的教师或者管理员实体，包括：用户信息，用户昵称，用户性别，通知标题，通知内容，通知作者 ID。

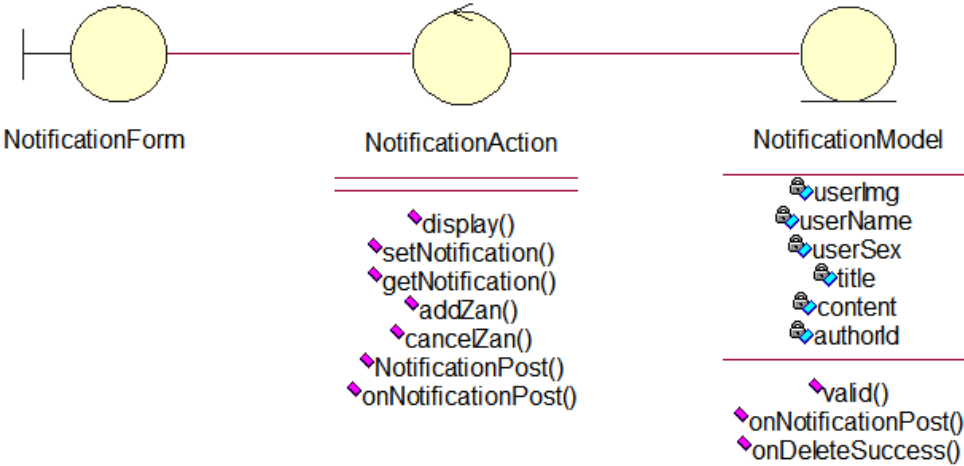


图 3-7 发布通知用例析取图

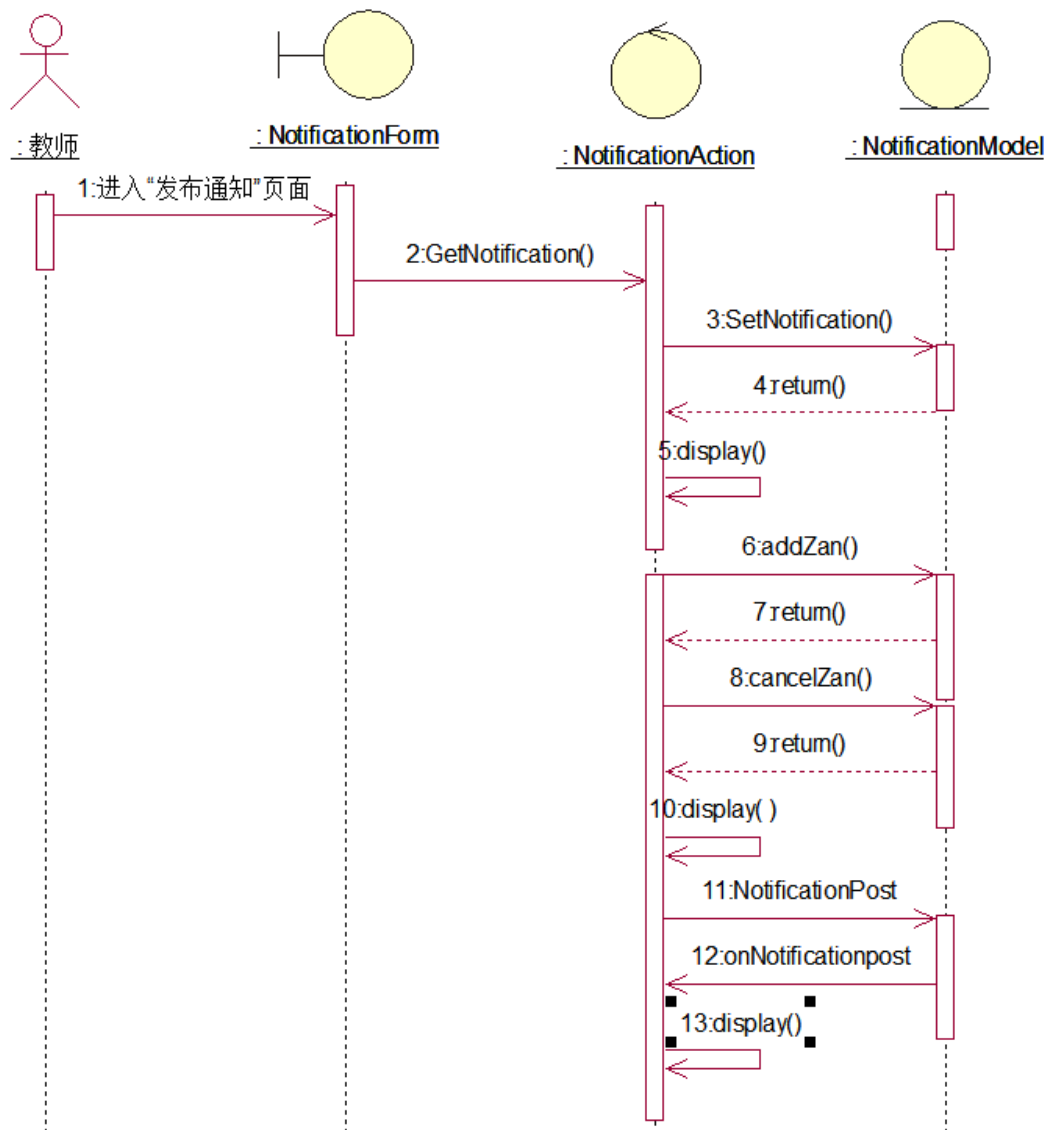


图 3-8 发布通知用例时序图

教师或管理员，点击课程详情中的发布按钮，通过 activity 跳转进入发布通知界面，输入通知的标题和内容，使用 `setNotification()` 编辑通知，用 `return()` 返回。`NotificationPost()` 发布通知，`onNotificationpost` 获取实体信息发布通知，`display()` 刷新显示。。

学生端、教师端进入通知界面，`getNotification()` 加载通知，使用 `addZan()` 来判断通知范围，`display()` 来显示通知。

3.2.5 选择课程用例析取

根据需求分析中的用例归约，经分析可得到该用例中的三种类。

边界类：courseForm 边界类为选择课程页面

控制类: **CourseAction** 控制类负责处理加入课程和退出课程的相关操作, 包括加入课程、退出课程、网络异常处理、用户关联课程、课程关联用户、更新课程列表、退出课程时删除用户与课程的关联和确认加入或退出课程。

实体类: **CourseModel** 实体类表示课程实体, 包含课程名、课程描述、上课时间、上课地点、课堂容量和加入课程邀请码。

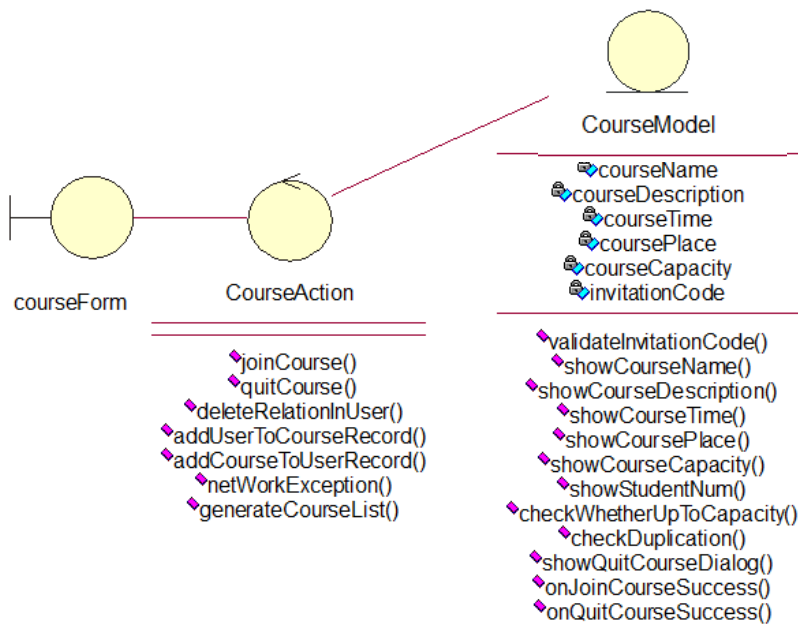


图 3-9 选择课程用例析取图

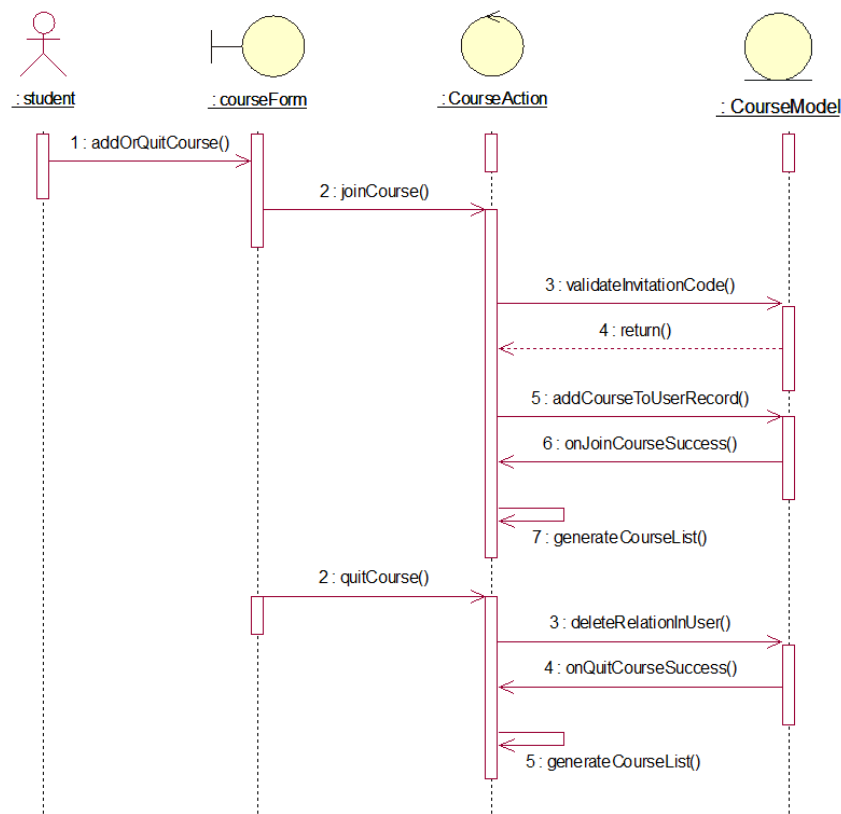


图 3-10 选择课程用例时序图

学生点击课程列表中的某一门课程，系统调用 `addOrQuitCourse()`方法进入加入或退出课程页面，点击加入课程调用 `joinCourse()`方法，退出课程调用 `quitCourse()`方法，加入课程后将课程加载到“我的课程”，退出课程则从“我的课程”删除该课程，最后调用 `generateCourseList()`更新“我的课程”课程列表。

3.2.6 课程维护用例析取

根据需求分析中的用例归约，经分析可得到该用例中的三种类。

边界类： `CourseMaintainForm` 边界类为课程维护页面

控制类： `CourseMaintainAction` 控制类负责处理课程维护的相关操作，包括删除、修改、添加课程信息，导出课程，更新课程列表。

实体类： `CourseModel` 实体类表示课程实体，包含课程名、课程描述、上课时间、上课地点、课堂容量和加入课程邀请码。

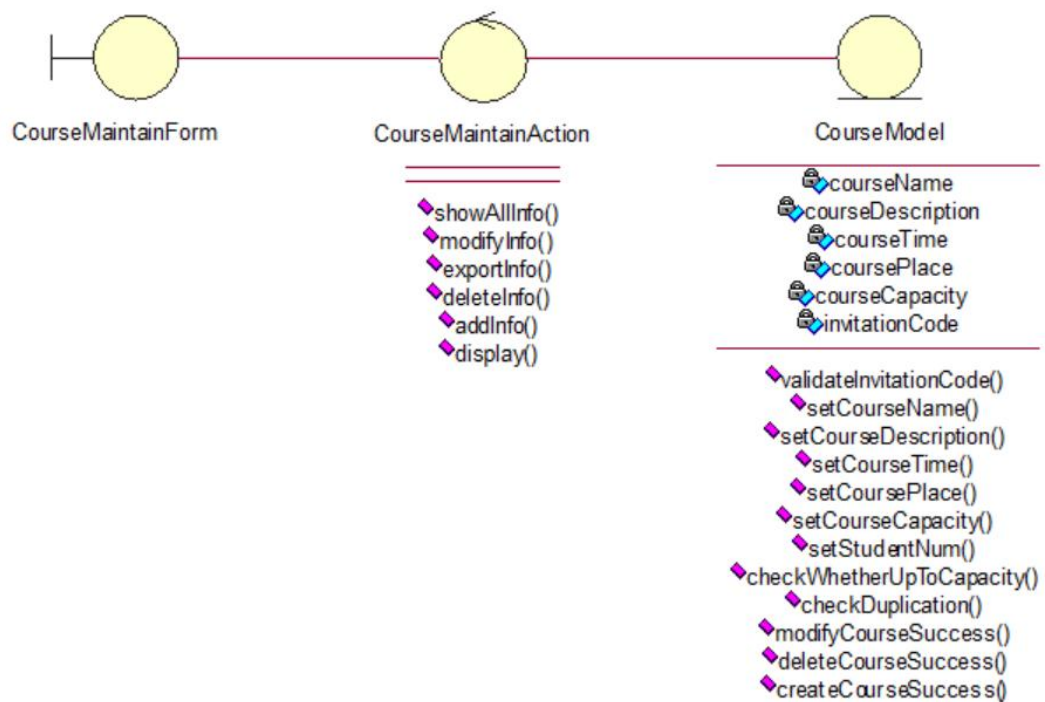


图 3-11 课程维护用例析取图

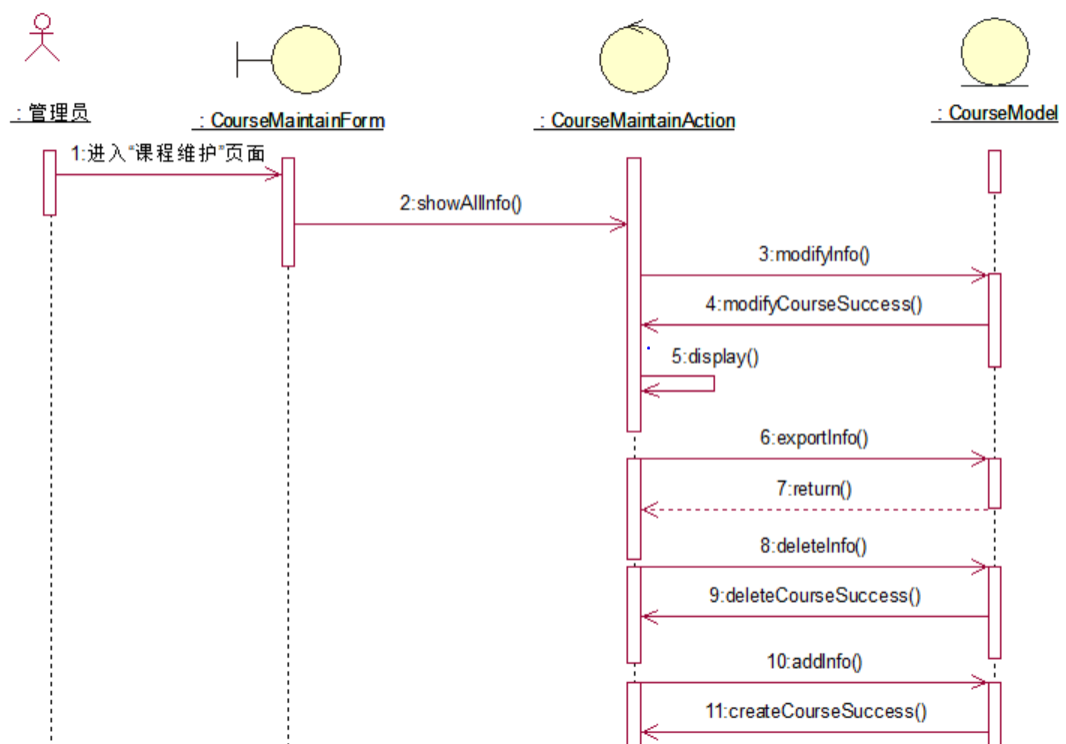


图 3-12 课程维护用例时序图

管理员进入“课程维护”页面，系统调用 `showAllInfo()` 显示所有课程，点击某一课程调用 `modifyInfo()` 方法修改课程信息，导出课程信息调用 `exportInfo()` 方法，添加课程信息

调用 `addInfo()`方法，删除课程调用 `deleteInfo()`方法，将课程从系统中删除，调用 `display()`更新课程列表。

3.2.7 人员管理用例析取

根据需求分析中的用例归约，经分析可得到该用例中的三种类

边界类： `UserManageForm`。 `UserManageForm` 是人员管理页面。

控制类： `UserManageAction`。 `UserManageAction` 控制类负责管理所有注册用户信息的相关操作，包括： 查看所有用户信息和资料、人员的信息的增添、删除或修改。

实体类： `UserModel`。 `UserModel` 实体类表示注册用户实体（被管理员继承），包含： 学号/工号，密码，邮件，管理员身份。

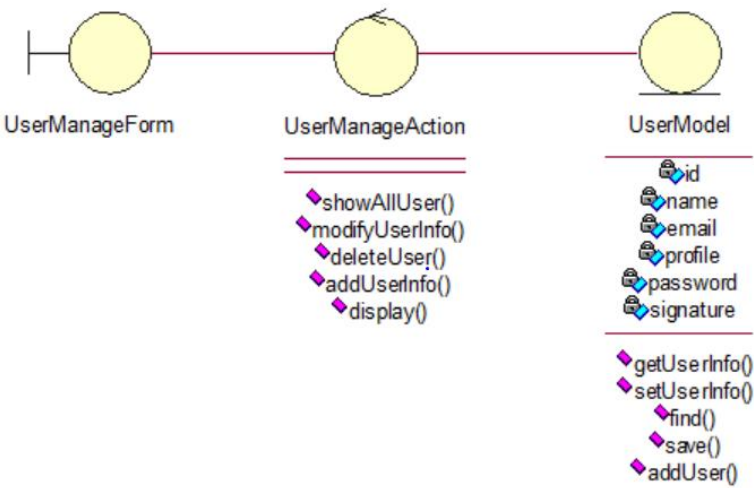


图 3-13 人员管理用例析取图

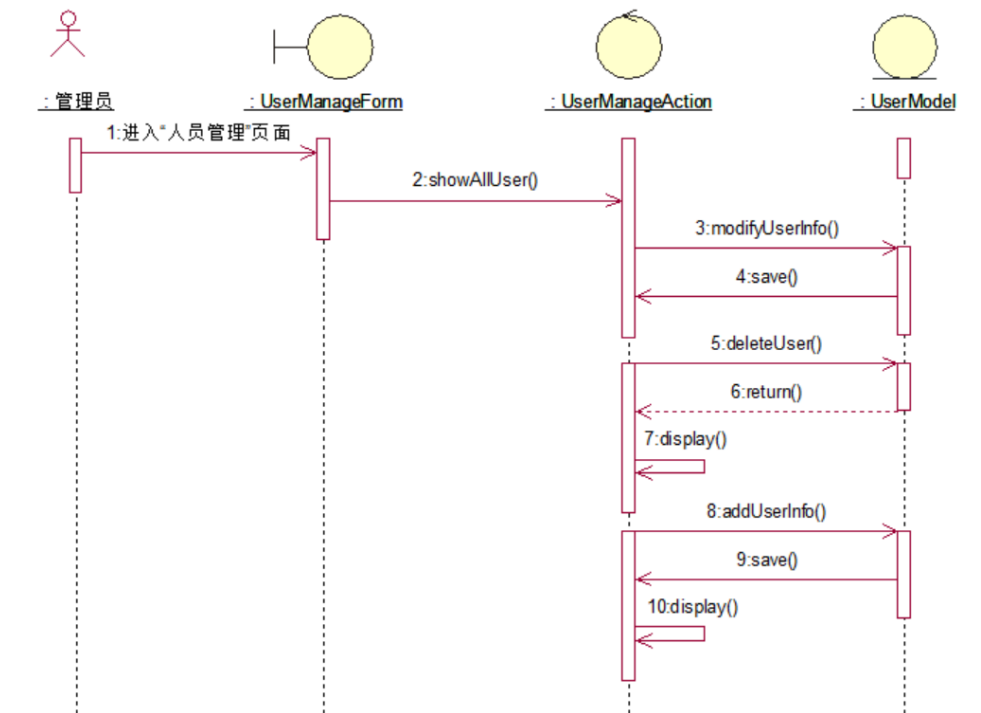


图 3-14 人员管理用例时序图

管理员进入“人员管理”页面，系统调用 `showAllUser()` 显示所有用户，点击某一用户调用 `modifyUserInfo()` 方法修改用户信息，添加用户信息调用 `addUserInfo()` 方法，删除用户调用 `deleteUser()` 方法，将用户从系统中删除，调用 `display()` 更新用户显示列表。

3.3 分析机制

分析类（analysis classes）	分析机制（analysis mechanism）
CommentModel	持久性、安全性
CourseModel	持久性、安全性
UserModel	持久性、安全性
NotificationModel	持久性、安全性

表 3-1 课程管理系统分析机制表

3.4 合并分析类

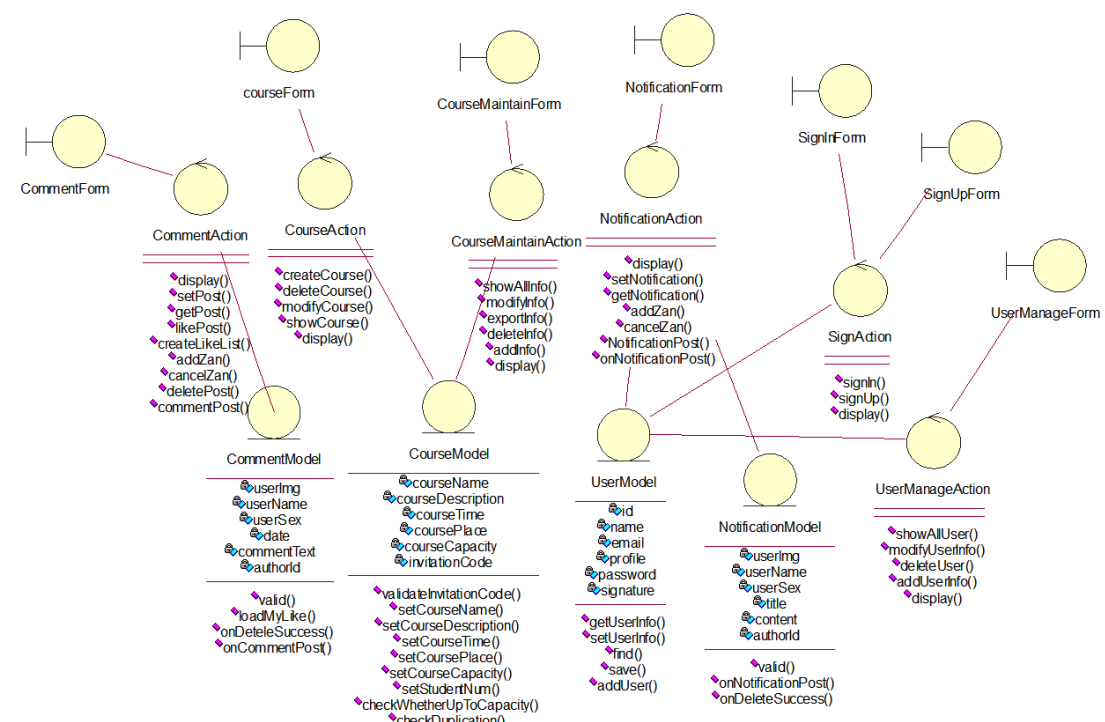


图 3-15 课程管理系统总类图

4. 子系统及其接口设计

4.1 确定设计类

本项目组对分析类进行了分析与检查，以确定其是否能成为设计类。经过分析发现，用例分析中所有分析类均为单逻辑，不需要进行类的分解或合并，因此不做修改，设计类如用例分析所示。

4.2 定义子系统

经过本项目组分析，本系统无需进行子系统设计。

4.3 定义接口

经过本项目组分析，本系统无需进行子系统接口设计。

4.4 确定可重用子系统

经过本项目组分析，本系统无需进行子系统设计及其接口设计，无可重用子系统。

5. 部件设计

类图中并未规定系统运行时功能，因此在系统运行时可能因为多线程多进程并发需求等产生资源抢占等问题。部件设计会分析系统运行时产生的一系列非功能性问题并给出解决方案。

5.1 分析并发需求

本系统在运行时会产生一些并发需求：

- (1) 多人同时使用该系统
- (2) 加入课程人数超过该课程的最大容量

5.2 针对加入课程人数问题的解决方案设计

系统运行时，课程加入人数可能会超过课程最大容纳量，此时可能会产生资源冲突问题。为了解决这个问题，我们使用了发布者设置加入课程邀请码机制，即加入课程人数可以超出课程最大容量，但只有持有发布者的课程邀请码的人可以加入课程。

5.3 生命周期

整个系统仅由一个进程构成，该进程生命周期与系统的生命周期一致，当系统结束运行时候，进程将结束；如果进程被迫结束（如发生异常），则该系统也会结束。

该进程包括多个线程。可能存在多个用户请求，需要一个请求线程，该线程仅用来接收所有用户的请求，并将请求放到对应的请求队列中去。需要一个响应线程对该队列进行轮询操作，如果发现请求队列存在用户的请求，则从资源池请求分配资源进行响应。处理线程由资源池进行分配，当响应线程请求一个线程处理请求的时候，如果资源池存在空闲的线程，则将该空闲线程用于处理该请求。

5.4 映射到现实系统

该系统仅适用于安卓手机用户，用户可通过下载安装 App，在联网的条件下能在现实系统中运行该设计方案。