

**<<KeepFit健身App>>**

**设计说明书**



北京航空航天大学

2015-12

版本变更历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 提交日期 | 主要编制人 | 审核人 | 版本说明 |
| A2015-00-02-00 | 2015/12/31 | 马元 | 邵璟璇 | 胡婧韬辅写 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目 录

[1. 范围 1](#_Toc20718)

[1.1 标识 1](#_Toc26021)

[1.2 系统概述 1](#_Toc16638)

[1.3 文档概述 1](#_Toc7142)

[1.4 术语和缩略词 2](#_Toc22538)

[2. 引用文档 2](#_Toc16889)

[3. 需求概述 2](#_Toc23116)

[3.1 用例整合 3](#_Toc15201)

[3.2 新增用例 4](#_Toc21671)

[4. 体系结构设计 5](#_Toc25403)

[4.1 总体结构 5](#_Toc14459)

[4.1.1 技术体系结构 5](#_Toc17717)

[4.1.2 软件体系结构 5](#_Toc2893)

[4.2 功能分配 11](#_Toc32547)

[4.3 关键问题及解决方案 11](#_Toc28790)

[5.接口设计 12](#_Toc6250)

[5.1用户界面设计 12](#_Toc17779)

[5.1.1典型的APP用户界面风格 12](#_Toc17819)

[5.2外部接口设计 14](#_Toc3468)

[5.3内部接口设计 14](#_Toc489)

[5. 数据结构设计 14](#_Toc4082)

[5.1 公共数据结构设计 14](#_Toc30057)

[5.2 数据库设计 14](#_Toc25471)

[6. 详细设计 16](#_Toc5692)

[6.1 用户模块 17](#_Toc15802)

[6.1.1 模块概述 17](#_Toc9231)

[6.1.2 内部结构设计 17](#_Toc18135)

[6.2 视频模块 21](#_Toc22678)

[6.2.1 模块概述 21](#_Toc707)

[6.2.2 内部结构设计 21](#_Toc15599)

[6.3 方案模块 22](#_Toc8060)

[6.3.1 模块概述 22](#_Toc21267)

[6.3.2 内部结构设计 22](#_Toc15175)

[6.4 闹钟模块 25](#_Toc31667)

[6.4.1 模块概述 25](#_Toc2436)

[6.4.2 内部结构设计 25](#_Toc1088)

[6.5 登陆模块 25](#_Toc13103)

[6.5.1 模块概述 25](#_Toc26092)

[6.5.2 内部结构设计 25](#_Toc30794)

# 范围

## 标识

适用系统：Android 5.0、Android 6.0

标识号：keepfit\_1.0

标题：KeepFit健身App

版本号：1.0

文档标识号：A2015-00-02-00

文档标题：系统设计说明书

版本号：1.0

发型号：1.0

## 系统概述

我们所开发的软件系统最主要的功能是根据用户需求和自身条件，提供相应的健身语音视频，记录用户的健身进展并在相应时间有闹钟提醒，用户可将健身进展和成效分享到软件自带或其他社交平台。同时，用户还包括发布健身视频的教练。软件系统将提供给用户第三种运动选择，把客厅和办公室变成健身房，把各种屏幕包括手机、电脑变成教练，在家里足不出户就可以完成健身。该系统应具有功能性、可靠性、易用性、高效性、维护性、可移植性。

本系统是我们项目开发小组开发的第一版软件，无历史版本。

运行现场？？？

## 文档概述

本文档的用途在于给出KeepFit健身App的体系结构、接口和数据结构设计，最后给出每个模块的详细设计。

文档的主要内容中，首先会对需求进行概述，补充新的细节，完善用例；接着用自顶向下的思路给出系统中主要模块的设计，将需求文档当中各项功能要求同总体结构对应，给出关键问题的设计决策和解决思路；接口设计部分给出用户界面，软硬件外部接口和系统内各模块接口的总体设计决策；数据结构设计部分给出公共数据结构和数据库的设计；最后的详细设计给出每一个模块（构件）的详细设计方案，直接指导编码人员进行开发。

本文档的预期读者是设计人员、开发人员、项目管理人员、测试人员，出于私密性和保密性考虑，除了这些人以外的其他人员都无权阅读该文档，只有设计人员经小组讨论后才能修改该文档。

## 术语和缩略词

无

# 引用文档

[1] KeepFit健身App项目开发计划书（1.0） 2015

[2] KeepFit健身系统需求规格说明书（1.0） 2015

[3] 《软件工程》 Roger S.Pressman 机械工业出版社 2015

# 需求概述

该系统非功能性需求中需要重点考虑的有安全性，防止用户非法输入；易恢复性，及时恢复数据库；可扩展性，类结构清晰；还有界面设计的易学习性，易用性。

功能性需求由用例图展现，以下是对用例图的变更

## 用例整合

各项功能需求在需求文档中已经表述为用例图，从中删除了重复的或可以合并的用例图，得到了要实现的用例图如下：

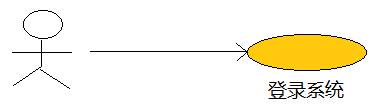


图 1：登录系统用例图

登陆用例将教练，锻炼者两种身份合并

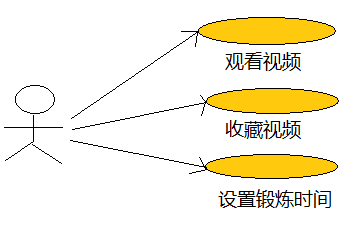


图 2：设置方案用例图

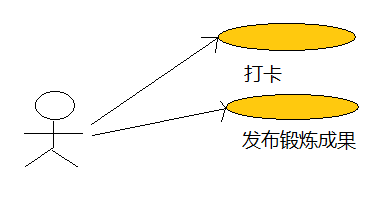


图 3：执行方案用例图

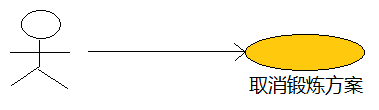


图 4：取消方案用例图

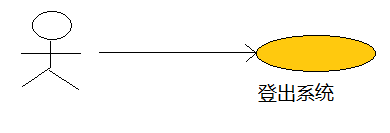


图 5：登出用例图

登出用例将教练，锻炼者两种身份合并

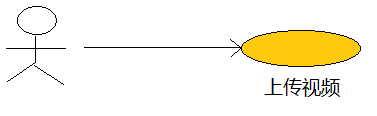


图 6：上传视频用例图

## 新增用例

对登出用例新增描述：

在基本路径中修改：

1. 当用户选择“登出系统”或者单击返回键时，用例开始
2. 若用户选择“登出系统”，系统询问用户是否登出
3. 用户选择“是”
4. 退出系统
5. 若用户单击了一次返回键，系统提示用户“再次单击退出应用”
6. 用户在3s内再次单击返回键
7. 系统退出

增加可选路径：5）后如果用户没有执行6），则系统保留在当前页面

# 体系结构设计

## 总体结构

### 技术体系结构

Android平台+Android内置sqlite数据库

### 软件体系结构

本系统采用面向对象分析与设计方法。依据需求规格说明书中的用例图设计功能模块和用例实现流程，以一张类图来描述所有模块的组成和关系，以一张活动图给出本系统最核心的业务逻辑，以六张时序图展现各个模块协作完成每个用例的过程，以一张状态图给出系统中较复杂的登出状态的变化情况。

1.模块类图：

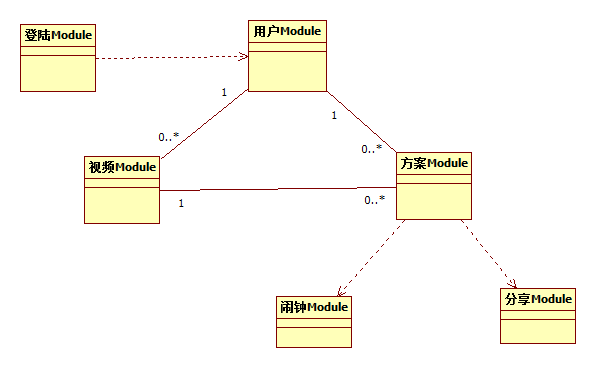


图 7：模块类图

2.用户锻炼活动图：

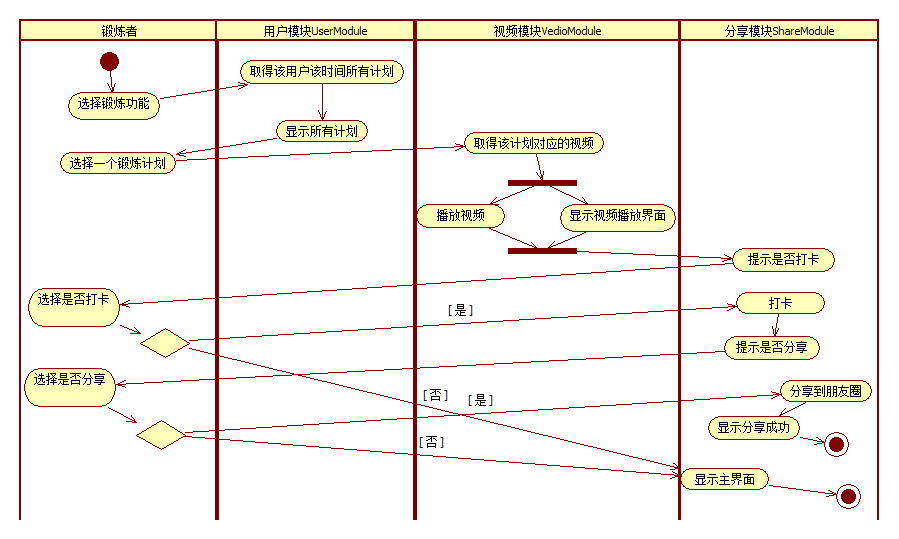


图 8：用户锻炼活动图

1. 登陆序列图：

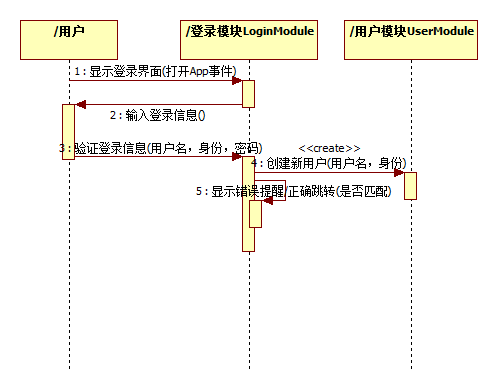


图 9：登陆序列图

1. 设置方案序列图

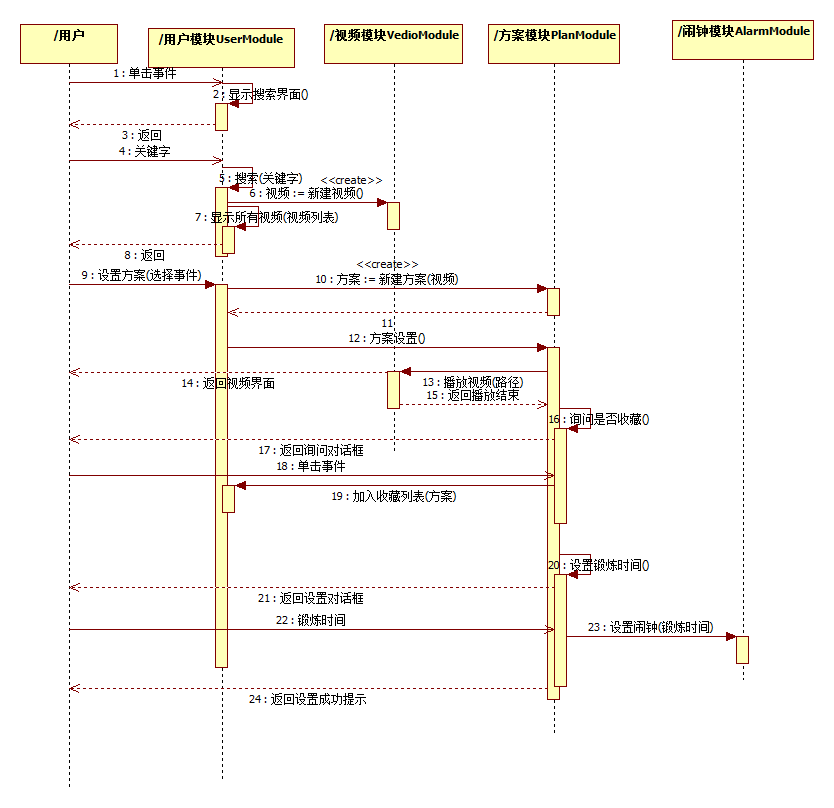


图 10：设置方案序列图

1. 执行方案序列图

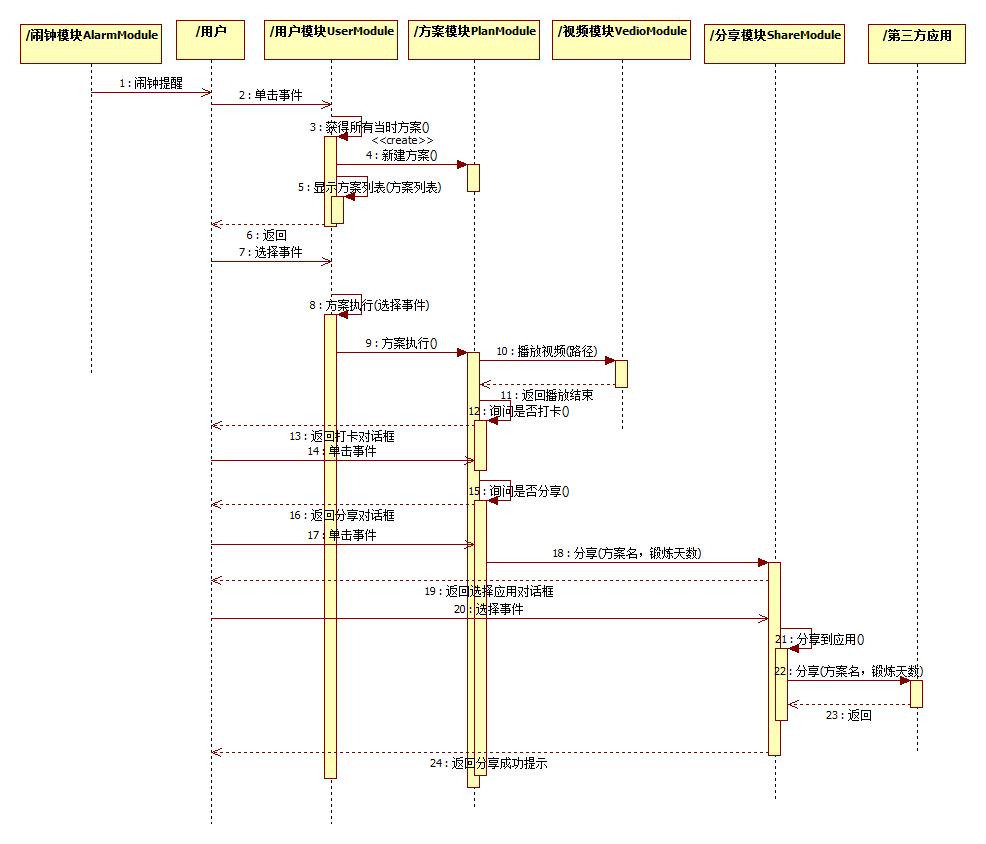


图 11：执行方案序列图

1. 取消方案序列图

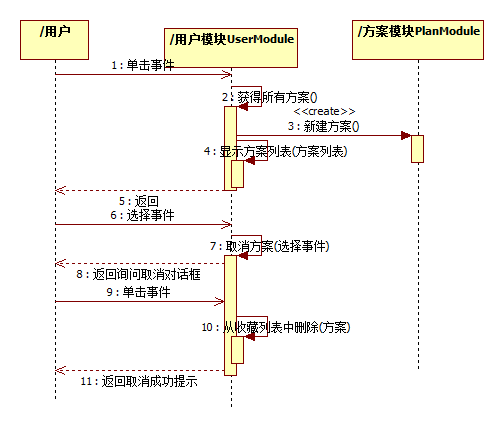


图 12：取消方案序列图

1. 登出序列图

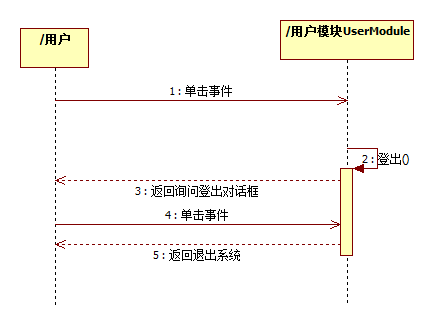


图 13：登出序列图

1. 上传视频序列图

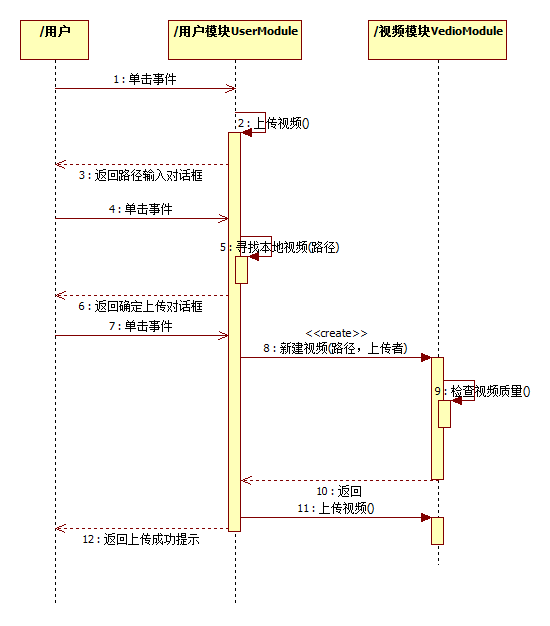


图 14：上传视频序列图

1. 系统状态图

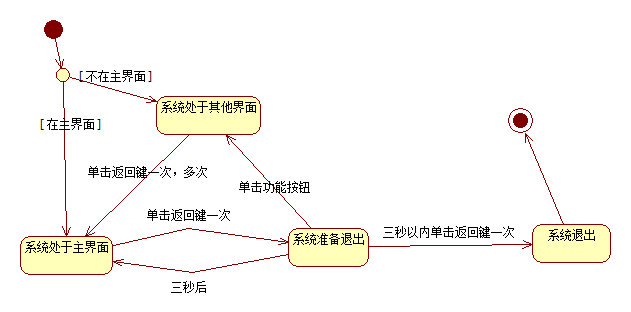


图 15：系统状态图

## 功能分配

各项功能需求在需求文档中已经表述为用例图，每个用例图都对应一个序列图，在序列图中可以看出各模块是如何协作完成功能需求的。该设计阶段以功能模块为粒度形成总体，每个模块的所有功能完全由用例决定，在序列图中就可以明确。

类的关系，模块分工明确满足了可扩展性。其他非功能需求在数据结构和接口设计中有所体现。

## 关键问题及解决方案

本次设计主要集中在软件体系结构，在这方面主要问题是模块和大致类的划分。最初的设计是全部按照功能划分，分为上传视频模块，设置锻炼计划模块等，在这些模块中完成所有特定的流程控制；所有向用户显示的部分，在Android里即Activity等也集合起来作为单独的显示模块；而把实体类功能架空，仅作为数据模板，提供最基本的操作接口，集合起来作为实体模块。这样设计类数目会增加很多，类功能过于单一，模块间交互会非常频繁，耦合性高，难以控制。

于是第二次设计以实体类为核心，在实体类提供的数据操作接口上封装控制流程和显示界面的功能，形成更接近用户需求的接口。即将大部分控制模块和显示模块合并到实体模块中，仅保留不可缺少的控制模块，如登录模块。这样类的数目和复杂度，模块间的耦合度都基本可以接受。不过功能分配仍稍有不均。

# 5.接口设计

## 5.1用户界面设计

## 5.1.1典型的APP用户界面风格

针对APP界面风格而言，主要是指APP通过主题色彩搭配、页面布局和NPC等给用户呈现出的整体视觉感受。

目前国内最流行的APP是微信和QQ，下文将对其界面风格的进行分析。



图 微信（左）和QQ（右）的用户界面

微信的主要颜色是白、黑、绿三种，给用户稳重、信赖的感觉。QQ默认皮肤的主要颜色是浅蓝、浅灰、白三种，灵活的布局交互呈现给用户活泼、有趣的感觉。这两种APP的设计风格差异主要出于产品定位和目标用户群的不同。根据资料显示，86%微信用户年龄集中在18至35岁，90%用户职业为企业员工、学生，目标用户也决定了微信设计风格的稳重、中性化和高端化；再观QQ，定位为娱乐化的社交应用，因此QQ设计风格是允许活泼有趣、个性化设置皮肤，是许多中小学生沟通交流的不二之选。

而两种APP的共同之处是其扁平化的设计风格，这也是当下的业界主流。这种扁平化设计的好处在于：界面美观、简约大方、条理清晰；设计元素上强调抽象、极简、符号化，去除冗余的装饰效果突显软件的文字图片等信息内容；完美兼容不同系统的平台和不同屏幕分辨率的设备，适应性强。

##### 5.1.2用户界面的总体设计决策

在设计风格表现上，颜色占据了80%以上的视觉体验。因此要做好设计风格，主要做好界面的颜色搭配和分布。另外颜色是有情感的，不同的色彩能给于用户不同的印象和感受，而且不同的人群对颜色偏好也是不一样的。所以在为APP设计进行配色时，需要考虑不同用户的喜好和体会配色给用户带来的视觉感受。



图 APP色彩设计分析

上图表示，男性和女性都倾向于绿色的蓝色，这也说明了为何典型的APP用户界面多以蓝色作为重要配色；男性和女性大多数讨厌橙色和褐色，但图表仅供参考，在团队前期考察时也发现，大多针对女性的APP采用粉色和红色作为主打色，而淘宝更是采用橙色作为主基调。

基于以上对于典型的APP用户界面风格分析，本组的美工团队进行了针对于健身APP的用户界面风格设计。主色调采用沁人心脾的浅蓝色、白色结合，给用户以舒适、愉悦、忘却烦恼的感受，更加有助于投入到之后的健身锻炼中。主界面以清晰的图片展示该项训练所塑形的身体部位或者主要动作，图片上附有简洁的训练说明，整体展现出符合潮流的扁平化视角。



对于界面设计的决策，团队还会及时进行用户满意度调查。根据当前的用户反馈情况，主界面的颜色对比度较小，给人感觉不是很清晰；该项目下一步更新会将浅蓝色加深，以适应更多用户的喜好，并且调整字体，给人以更加活泼向上的视觉体验。

## 5.2外部接口设计

基于KeepFit健身APP的分享模块，本项目的外部接口有用户手机的相册、相机、以及微信、微博等社交平台。这些接口均有Android平台提供。

## 5.3内部接口设计

显示界面的跳转时如果附加数据，均使用Bundle

数据库返回的每一行数据使用HashMap<String,Object>

所有集合的传递使用ArrayList<Object>

# 数据结构设计

## 公共数据结构设计

Pulic数据项都在数据库模块中，如数据库中字段名，字段的常量取值定义为public static final String

## 数据库设计

每张表要根据值域设置完整性约束，程序中的数据库模块中在接收任何输入数据时都要根据值域进行检查，做错误处理，以保证安全性。数据库还应设置定时转储机制，在发生错误后及时恢复，这一点暂时没有具体设计。

数据库相关模块和类都没有在体系结构设计和详细设计中体现，因为它复用性很高，可以很好地独立于具体软件的功能性需求。具体考虑使用一个DBHelper类，每次调用子类DBOpenHelper（使用了单例模式，所以不会反复创建）创建，打开数据库，提供打开，关闭数据库，对特定表进行增删改查的接口。所有类在需要使用时只需创建一个DBHelper的实例即可。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据库的表结构 | 数据库名称 | 表名称 | 设计者 | 审核者 | 完成日期 |
| keepfitDB | Exerciser | 马元 | 邵璟璇 | 2015/12/29 |
| Coach |
| Video |
| Plan |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 数据项名 | 含义 | 类型 | 值域 | 索引/键 | 备注 | 属于表 |
| e\_id | 锻炼者序号 | int | 正整数 | 主键，聚集索引 | 自动增长 | Exerciser |
| e\_name | 锻炼者登录的账号 | text | 字母数字下划线组成，长度<=10 | 索引 |  |
| e\_key | 锻炼者登录密码 | text | 字母数字组成，长度<=10 |  |  |
|  | | | | | | |
| c\_id | 教练序号 | int | 正整数 | 主键，聚集索引 | 自动增长 | Coach |
| c\_name | 教练登录的账号 | text | 字母数字下划线组成，长度<=10 | 索引 |  |
| c\_key | 教练登录密码 | text | 字母数字组成，长度<=10 |  |  |
| c\_sex | 教练性别 | text | {男，女} |  |  |
| c\_age | 教练年龄 | int | 3-99 |  |  |
| c\_degree | 教练资质等级,数值越大，等级越高 | int | {1，2，3，4，5} |  |  |
|  | | | | | | |
| v\_id | 视频序号 | int | 正整数 | 主键，聚集索引 | 自动增长 | Video |
| v\_name | 视频名称 | text | 字母数字下划线组成，长度<=10 |  |  |
| v\_cid | 视频上传教练序号 | text | 正整数,存在的c\_id | 外键，引用Coach表c\_id，索引 |  |
| v\_path | 视频路径（带扩展名） | text | 合法路径 |  |  |
|  | | | | | | |
| 数据项名 | 含义 | 类型 | 值域 | 索引/键 | 备注 | 属于表 |
| p\_id | 方案序号 | int | 正整数 | 主键，聚集索引 | 自动增长 | Plan |
| p\_eid | 方案中锻炼者序号 | int | 正整数，存在的e\_id | 外键，引用Exerciser  表e\_id，索引 |  |
| p\_vid | 方案中视频序号 | int | 正整数，存在的v\_id | 外键，引用Video表v\_id |  |
| p\_alarm | 方案提醒时间 | text | 时间格式“08：09”，合法时间，精确到分钟 |  | 24小时制 |
| p\_day | 方案已锻炼天数 | int | 非负整数 |  | 单位是天 |

# 详细设计

体系结构设计中给出的有些模块因为功能过于单一在本阶段被整合到其他模块，这里不再单独列出。本阶段给出的序列图是以类为粒度对上阶段的序列图，状态机图做出的细化。以下为系统类图：

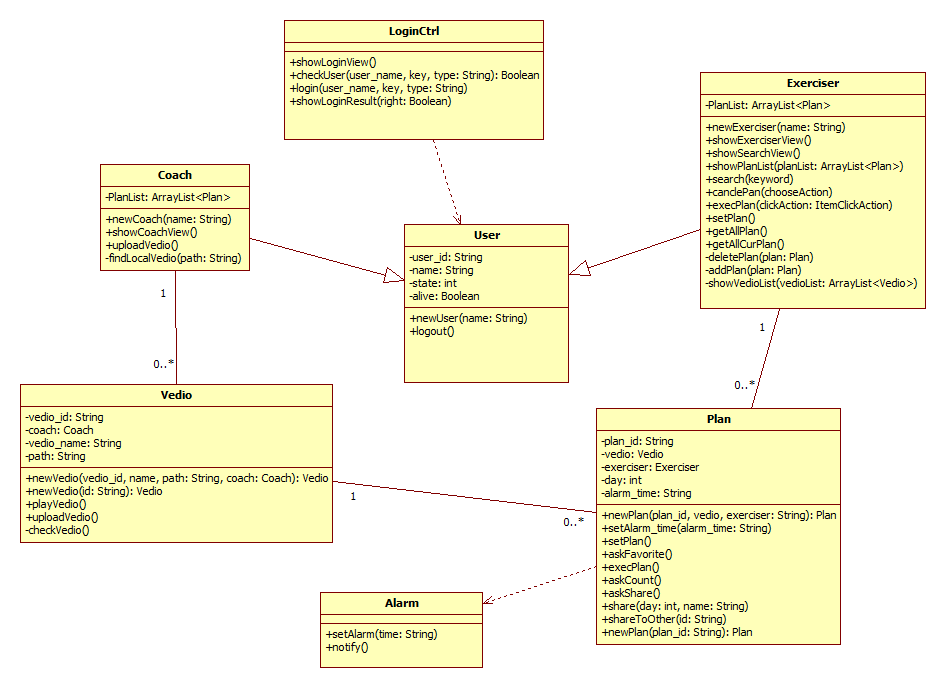


图16 系统类图

## 用户模块

### 模块概述

功能：用户主界面显示，完成用户功能的流程控制，锻炼者提供管理已设置方案的接口，教练提供管理已上传视频的接口，管理用户的详细信息

### 内部结构设计

静态结构：分为三个类实现，锻炼者和教练类继承于用户类，这种关系表述在上文中的设计类图中。以下为三个类分别的类图：



动态结构：

用户类作为父类以其独有的方法参与的业务流程较少，用于登陆时锻炼者、教练的创建和登出系统。由于登出时逻辑稍显复杂，序列图无法清晰表述，所以使用一张状态图来表示用户类状态变化具体逻辑。

教练类主要参与登陆和上传视频的业务。

锻炼者类涉及的业务流程较多，登陆，设置，执行，取消方案都有，逻辑比较多。

以下为登出系统，取消方案，上传视频序列图，用户类状态图：

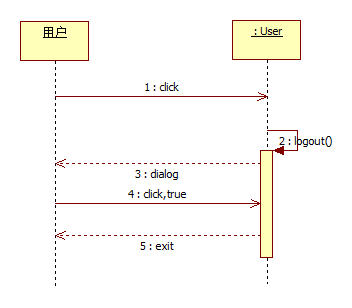


图17 登陆序列图

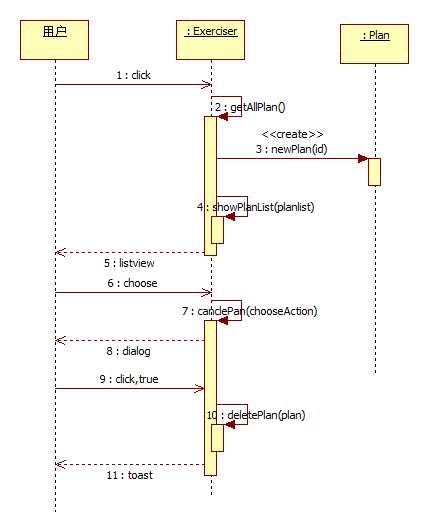


图18 取消方案序列图

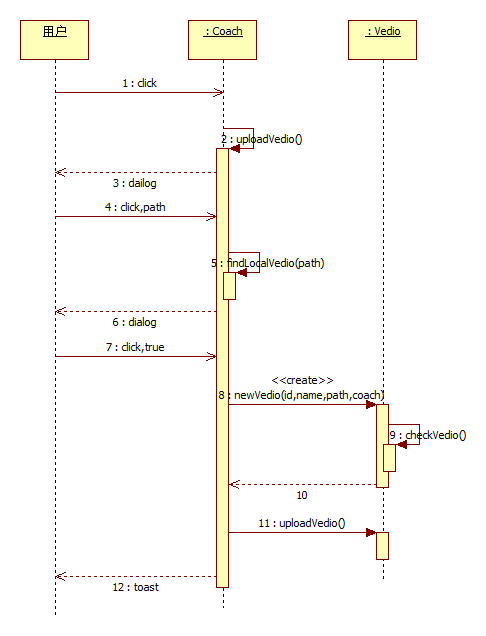


图19 上传视频序列图

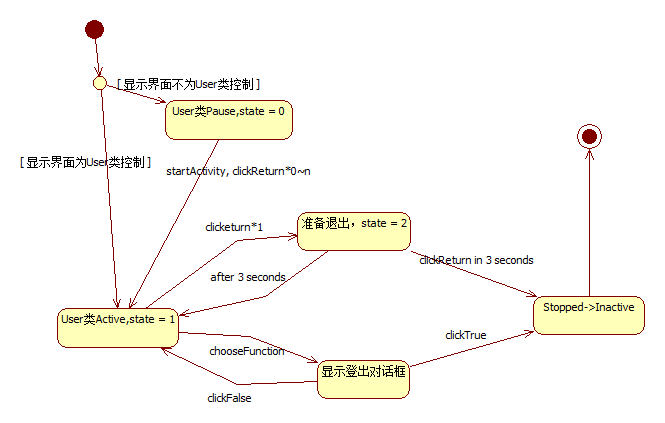


图20 User类登出状态图

## 视频模块

### 模块概述

功能：视频播放及控制，视频质量检查逻辑，视频上传至数据库的流程控制和实现，管理视频的详细信息

### 内部结构设计

静态结构：由一个类实现



动态结构：需要播放，上传视频的业务流程都需要参与，可参考其它模块中给出的设置，执行方案，上传视频序列图

## 方案模块

### 模块概述

功能：方案设置，方案更新，打卡，分享的流程控制与实现，管理方案的详细信息

### 内部结构设计

静态结构：由一个类实现



动态结构：设置和执行方案两个业务中起到关键作用，下面是两个流程的序列图：

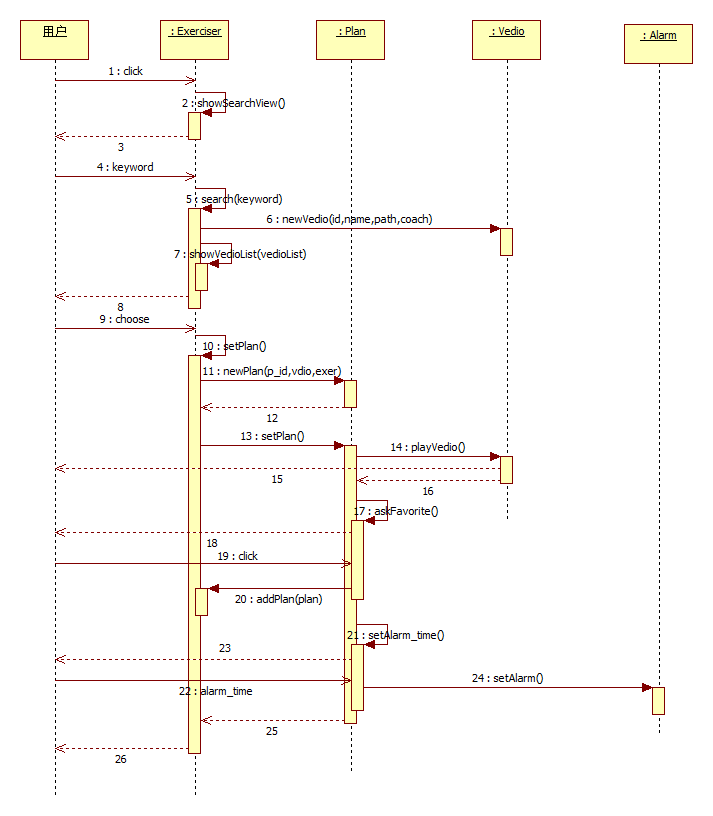


图21 设置方案序列图

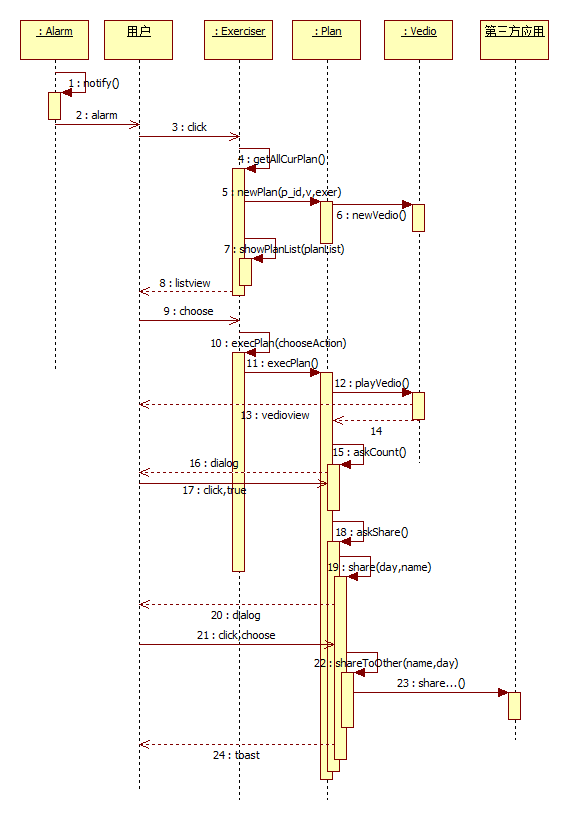


图22 执行方案序列图

## 闹钟模块

### 模块概述

功能：设置闹钟，接收系统广播到时间提醒，显示提醒界面

### 内部结构设计

静态结构：由一个类实现



动态结构：参见设置和执行方案两个业务的序列图，具体闹钟功能的实现参照普遍的模板。

## 登陆模块

### 模块概述

功能：显示登陆界面，控制登录流程，建立每次运行时的用户类实例

### 内部结构设计

静态结构：由一个类实现



动态结构：登陆业务中主要控制，下面为登录系统流程图：

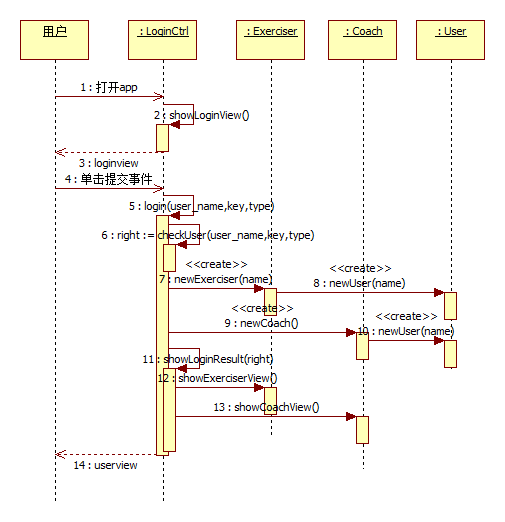


图23 登陆序列图