**软件分析设计文档**

**<Okane-Memo>**

**修订记录**

|  |  |  |
| --- | --- | --- |
| **修改人** | **修改时间** | **修改内容** |
| 林敏怡 | 2021/10/20 | 创建项目章程文档 |
| 林敏怡 | 2021/10/24 | 完善文档内容 |

目录

[1. 简介 5](#_Toc88299869)

[1.1 目的 5](#_Toc88299870)

[1.2 范围 5](#_Toc88299871)

[1.3 参考资料 5](#_Toc88299872)

[1.4 概述 5](#_Toc88299873)

[2. 系统分析模型 5](#_Toc88299874)

[2.1 用例模型 5](#_Toc88299875)

[2.2 领域模型 6](#_Toc88299876)

[2.3 系统顺序图 6](#_Toc88299877)

[2.3.1 查看数据 6](#_Toc88299878)

[2.3.2 记录支出/收入 7](#_Toc88299879)

[2.3.3 导出数据 7](#_Toc88299880)

[2.3.4 同步数据 8](#_Toc88299881)

[3. 系统设计模型 8](#_Toc88299882)

[3.1 体系架构表示 8](#_Toc88299883)

[3.2 体系架构目标和约束 8](#_Toc88299884)

[3.3 用例视图 9](#_Toc88299885)

[3.4 逻辑视图 10](#_Toc88299886)

[3.4.1 概述 10](#_Toc88299887)

[3.4.2 重要体系结构设计包 10](#_Toc88299888)

[3.4.3 用例实现 11](#_Toc88299889)

[3.4.3.1 类图 11](#_Toc88299890)

[3.4.3.2 查看数据 11](#_Toc88299891)

[3.4.3.3 记录收入/支出 13](#_Toc88299892)

[3.5 流程视图 14](#_Toc88299893)

[3.5.1 查看数据 14](#_Toc88299894)

[3.5.2 记录支出/收入 15](#_Toc88299895)

[3.6 部署视图 15](#_Toc88299896)

[3.7 实施视图 16](#_Toc88299897)

[3.7.1 概述 16](#_Toc88299898)

[3.7.2 层 16](#_Toc88299899)

[3.8 数据视图 17](#_Toc88299900)

[3.8.1 ER图 17](#_Toc88299901)

[3.8.2 系统数据字典 17](#_Toc88299902)

[3.9 质量 18](#_Toc88299903)

文档目的

本文档资料的目的是用来确定适合本项目的政策、标准及程序。文档资料也阐明了何时、何人如何使用这些政策、标准及程序。

如未有特殊申明，本文档资料的内容适合于项目组中的所有人。

# 简介

## 目的

本文档对Okane-memo记账本APP进行分析和设计。使用领域模型、用例模型、系统顺序图等描述软件的分析过程，并使用许多不同的体系结构视图来描述系统的多个方面，对系统进行了全面的体系结构概述。其目的是捕获和传达对系统作出的重要体系结构决策。

## 范围

本文档概述了Okane-memo记账本APP的分析设计过程和体系结构。该APP由林敏怡、徐越、詹熹共同开发。该APP基于Java开发，可运行在Android系统上。

## 参考资料

[1]（美）CraigLarman，UML和模式应用（第三版），机械工业出版社

## 概述

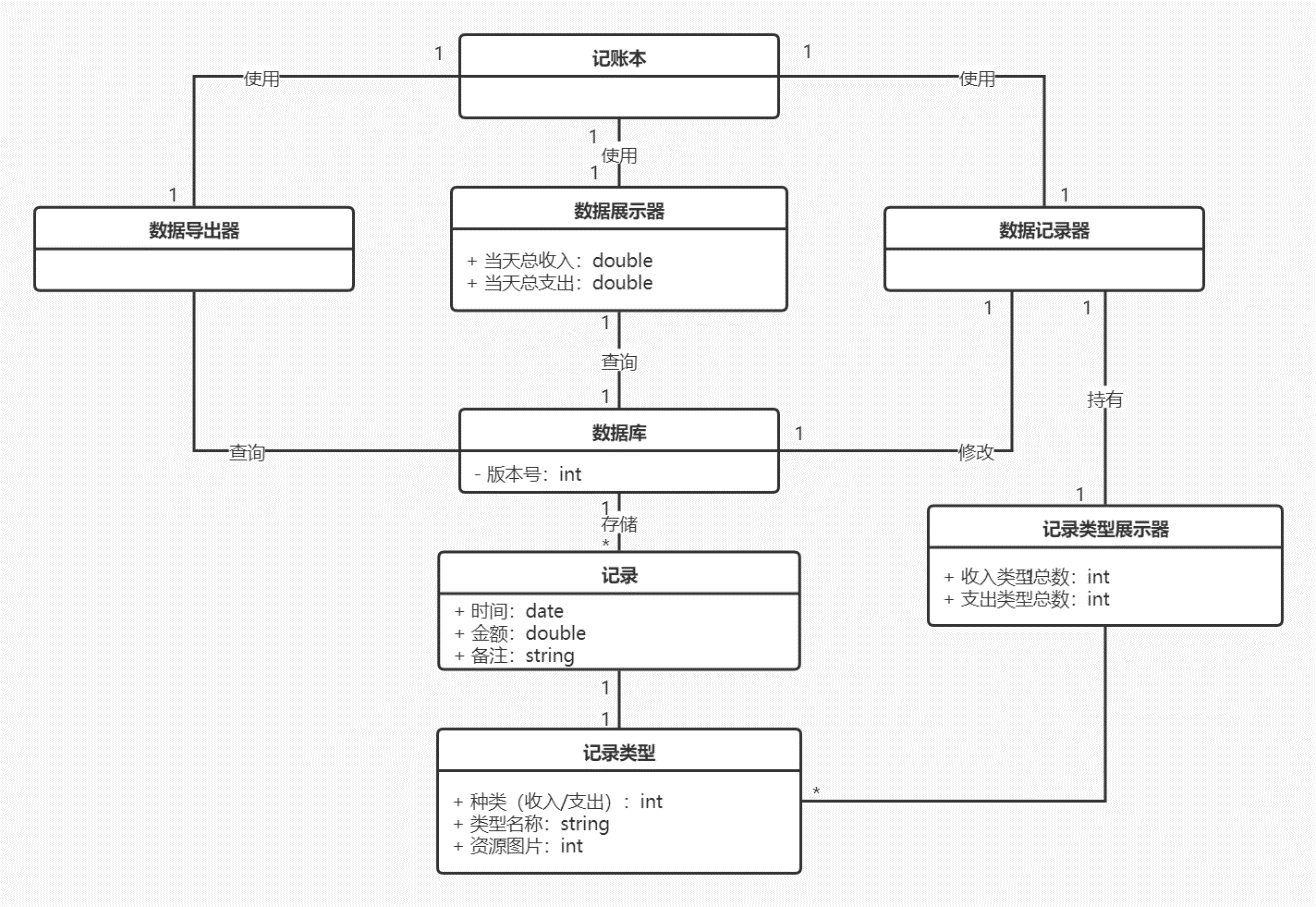
该文档包含简介、软件分析、体系结构、N+1视图等主要部分。

# 系统分析模型

## 用例模型

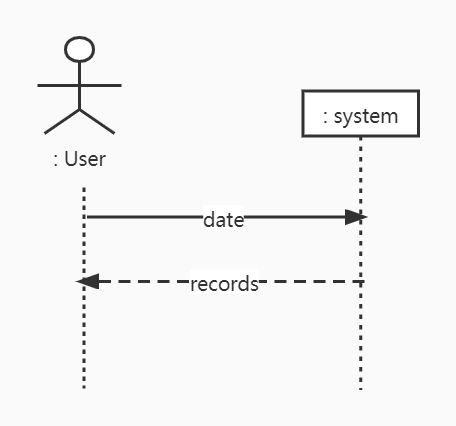
详见用例模型文件夹下的文件：用例模型.pdf

## 领域模型

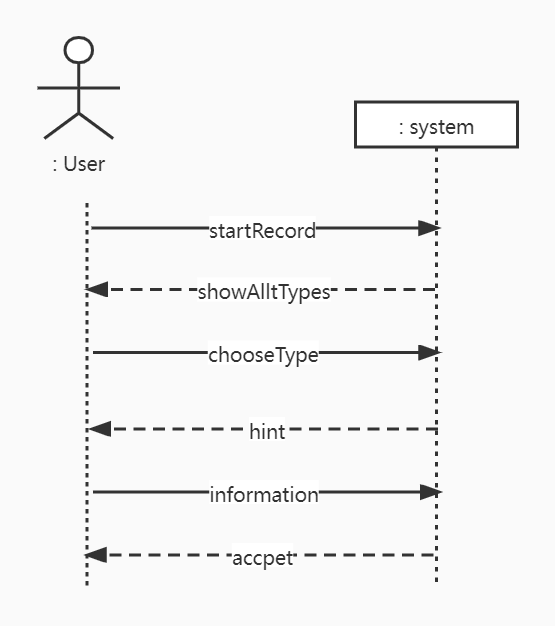


## 系统顺序图

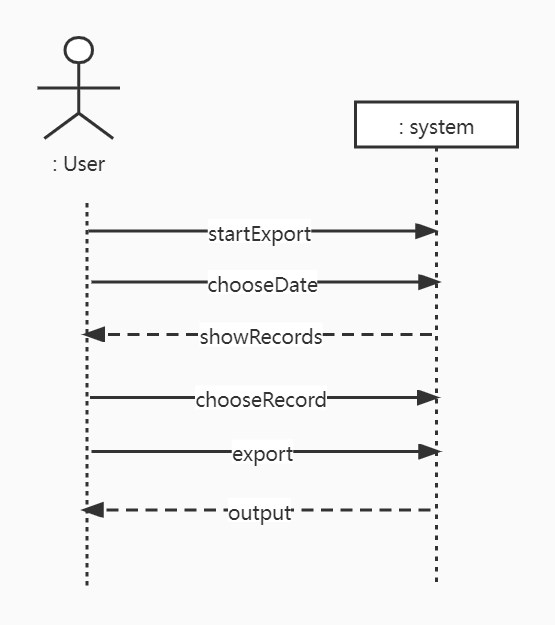
### 查看数据



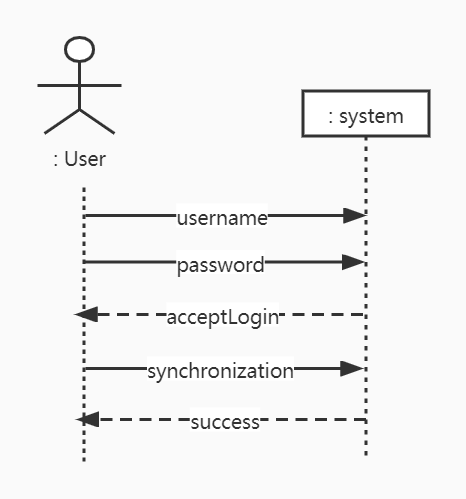
### 记录支出/收入



### 导出数据



### 同步数据



# 系统设计模型

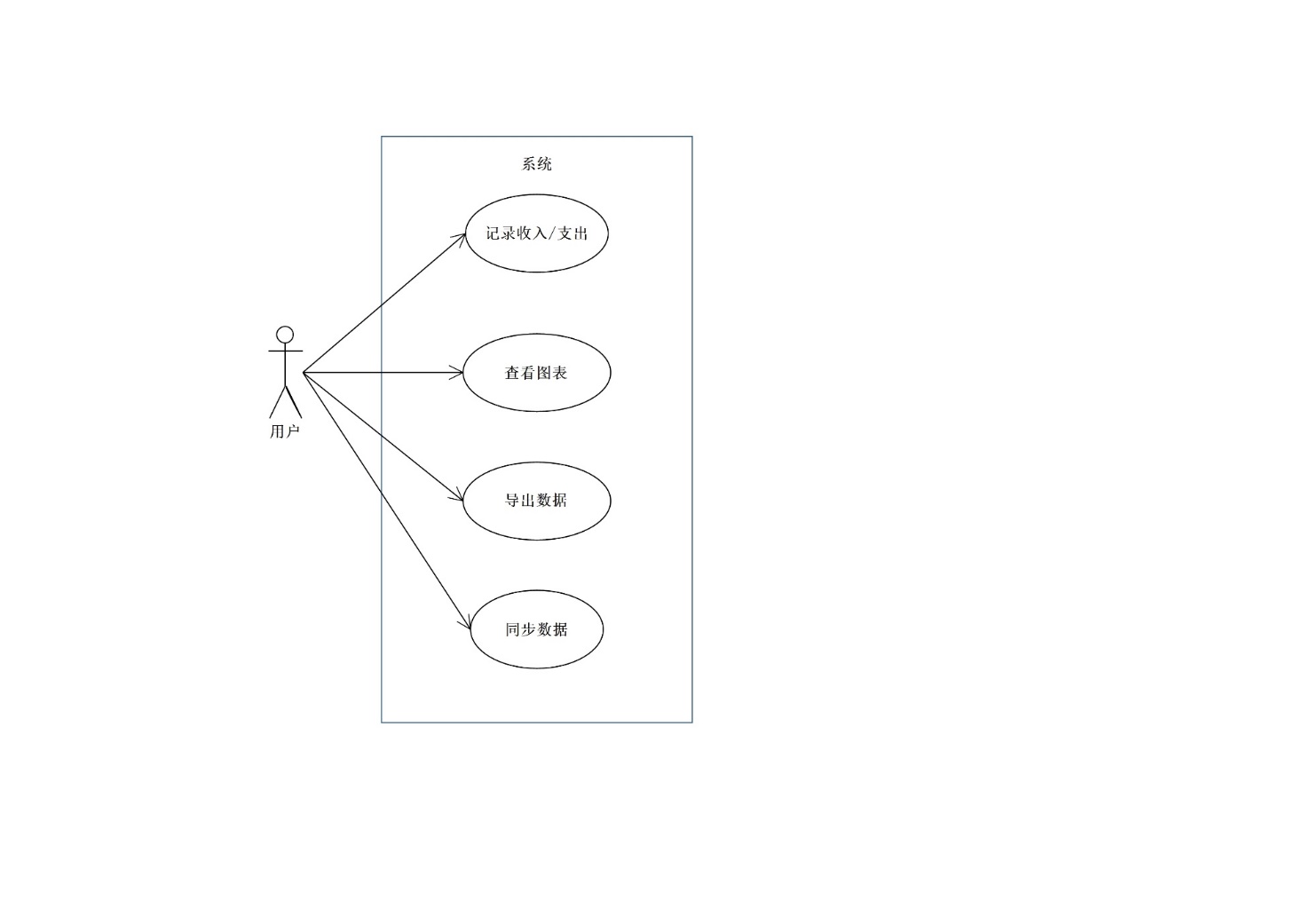
## 体系架构表示

本文档将体系结构表示成一组视图：逻辑视图、流程视图和部署视图。 但不描述单独的实施视图。

## 体系架构目标和约束

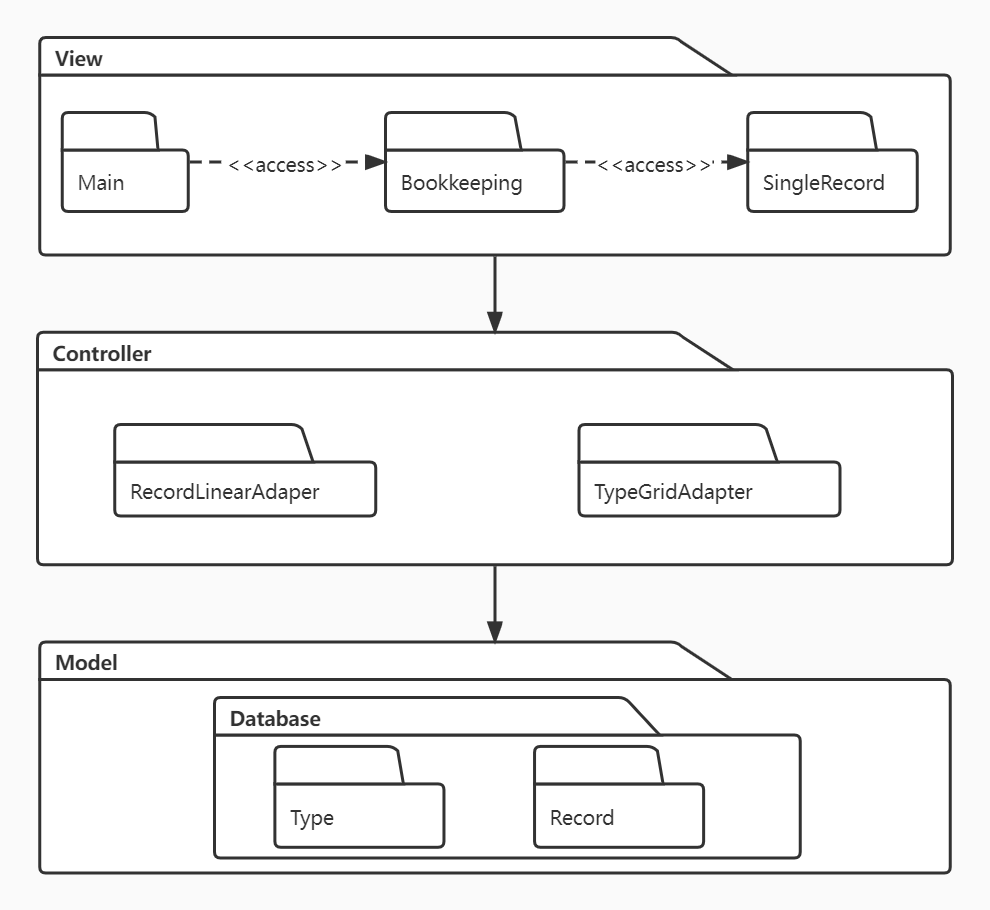
该系统采用低耦合、高类聚的方式进行设计，实现了高可用性、易用性、互操作性和安全性。

## 用例视图



## 逻辑视图

### 概述

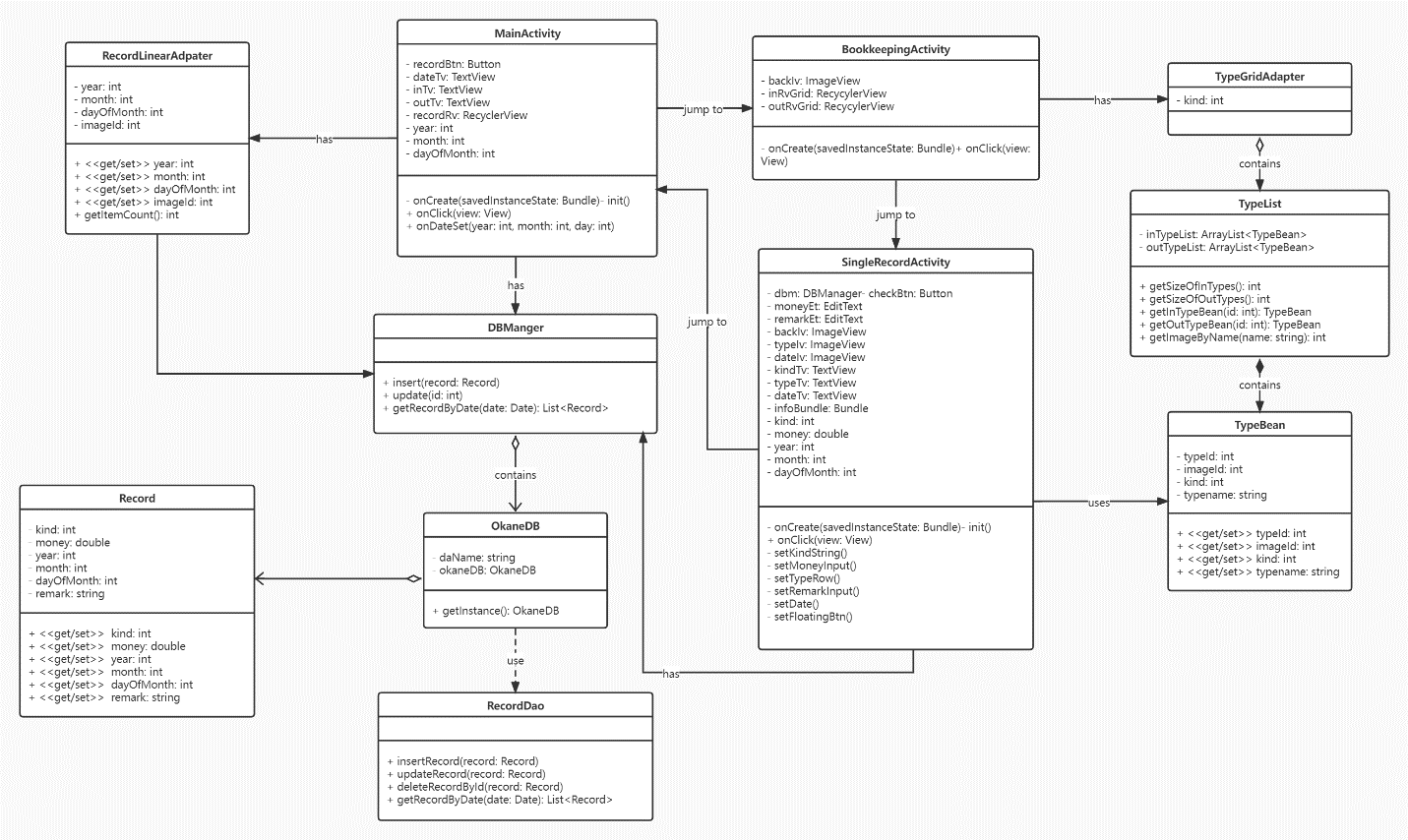


### 重要体系结构设计包

1. View包：主存存放于前端布局相关的文件
2. Controller包：将数据模型填充进View中
3. Model包：定义数据，存储数据

### 用例实现

### 类图

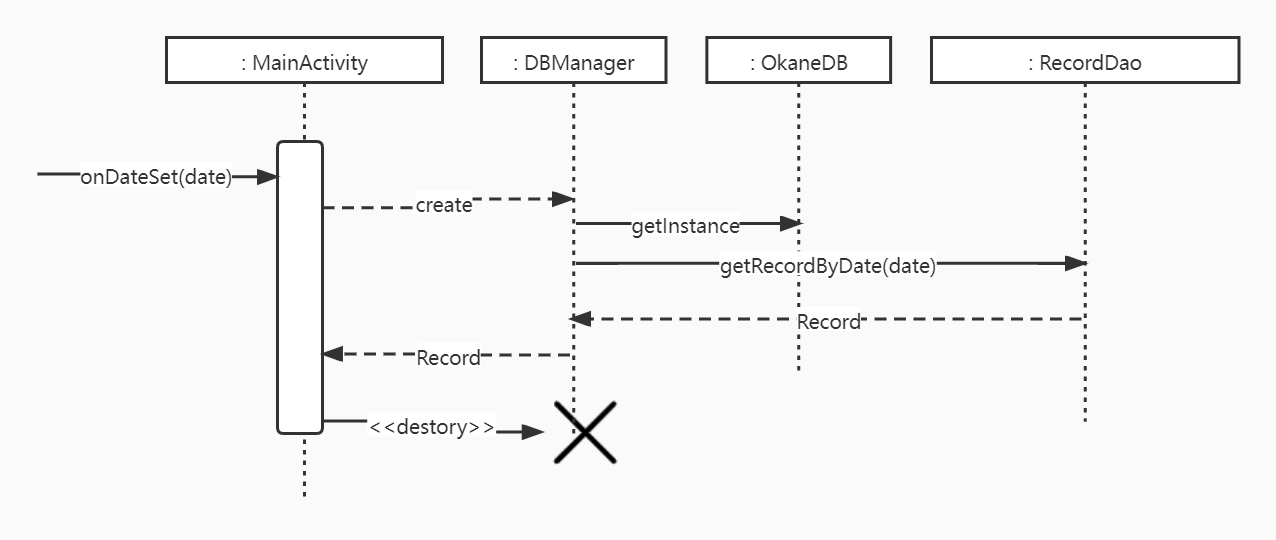


### 查看数据

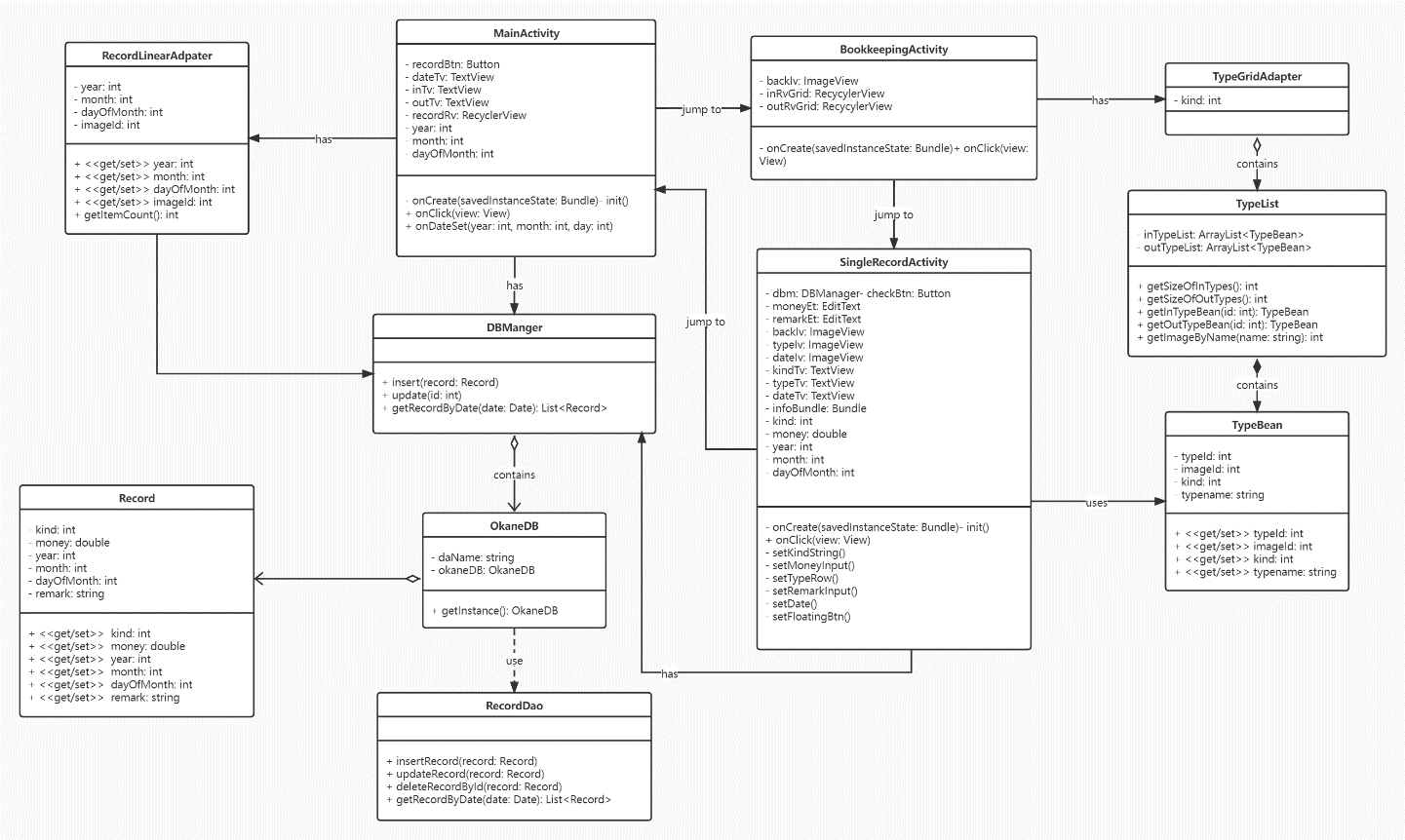
1. 简述

该用例描述用户查看特定日期数据的流程

1. 事件流
2. 用户选择日期
3. 系统从数据库中找出所有对应的记录
4. 系统显示数据
5. 交互图



1. 参与对象
2. MainActivity：主界面对象类，用于展示数据
3. DBManager： 数据库管理类，用于获取数据库实例并使用
4. OkaneDB:：数据库类
5. RecordDao：数据库访问类，用于增删改查数据库中的数据
6. 类图

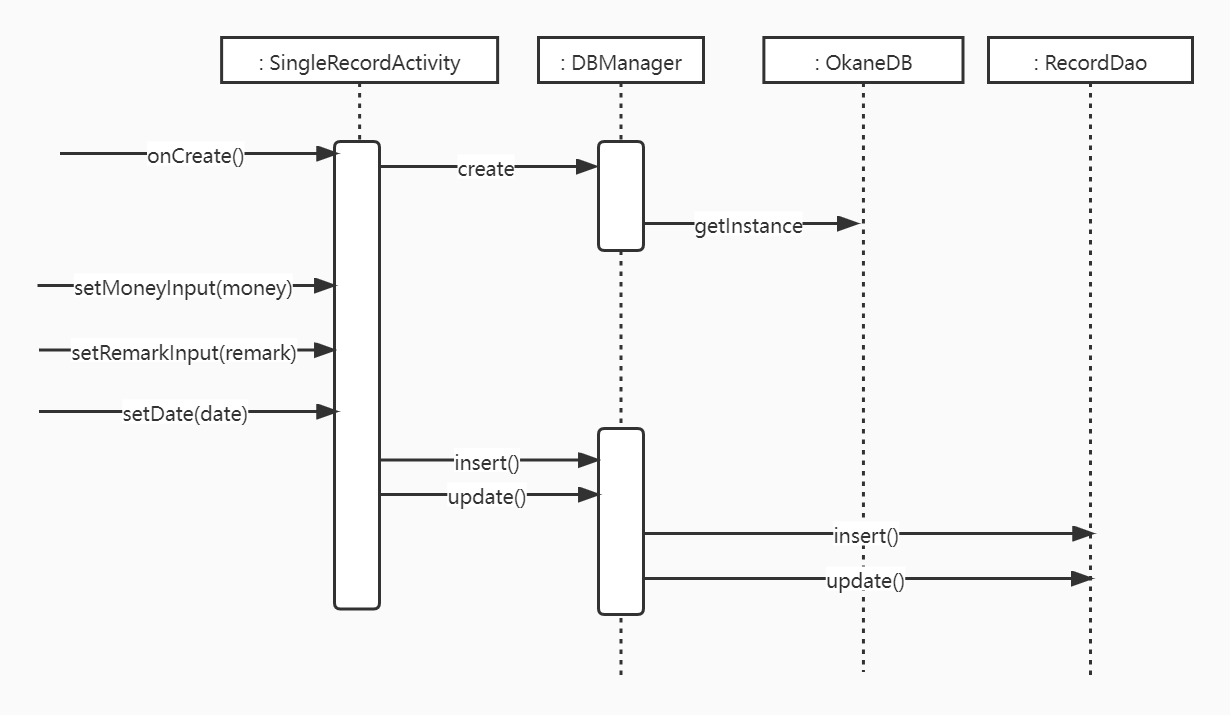


### 记录收入/支出

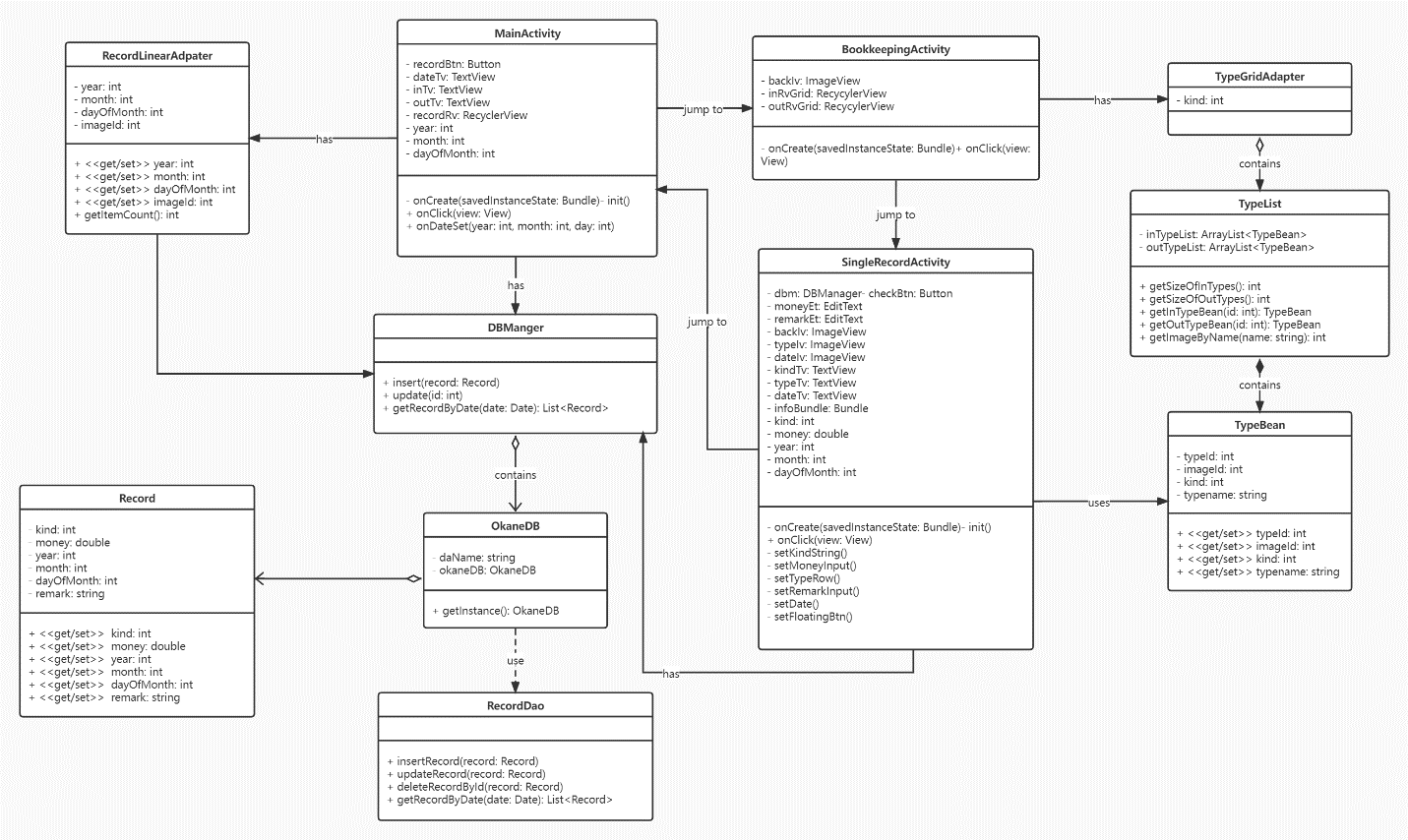
1. 简述

该用例描述用户记录收入/支出数据的流程

1. 事件流
2. 用户进入应用程序主界面，点击“+”浮动按钮
3. 跳转到添加账目记录页面
4. 用户通过左右横滑选择记录收入或支出
5. 用户根据记录账目类别点击响应图标
6. 跳转到账目详细编辑页面
7. 用户输入支出/收入的金额数
8. 用户输入备注
9. 用户确认相关信息后点击“√”浮动按钮，新增记录成功。
10. 界面底部显示“新增成功”的提示信息
11. 应用程序跳转到主界面
12. 主界面刷新并展示账目信息。
13. 交互图

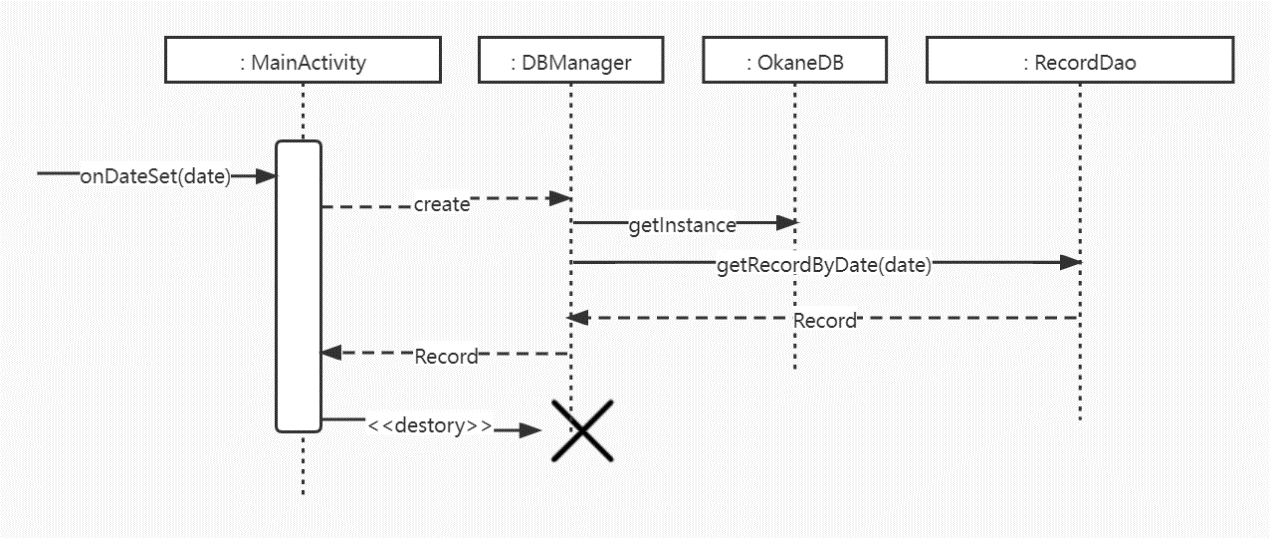


1. 类图

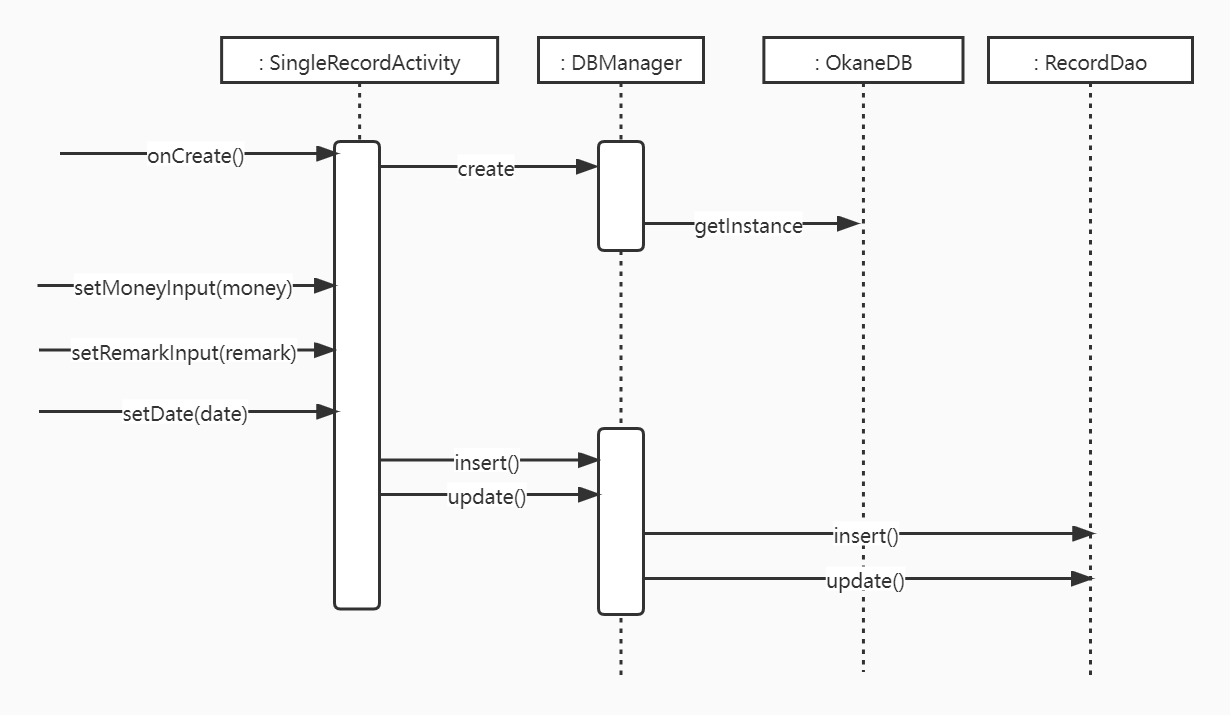


## 流程视图

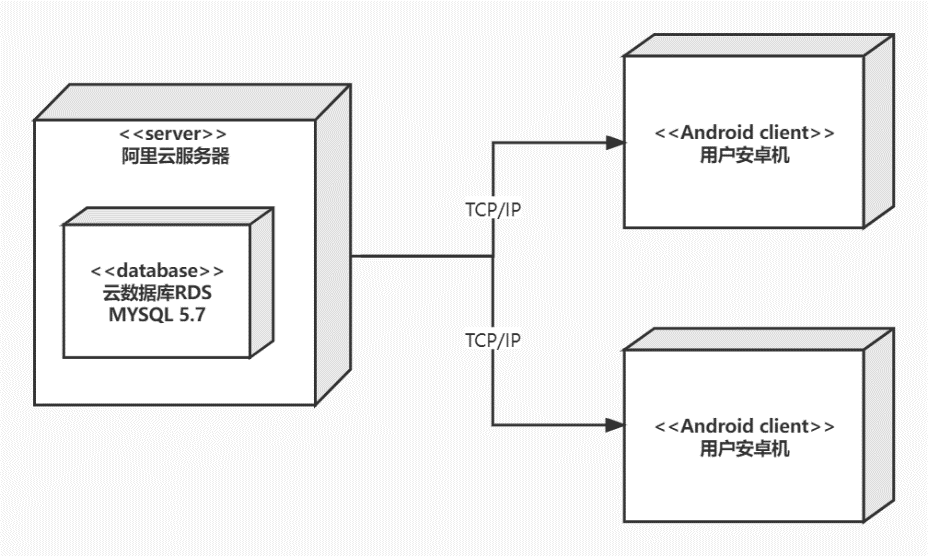
### 查看数据



### 记录支出/收入

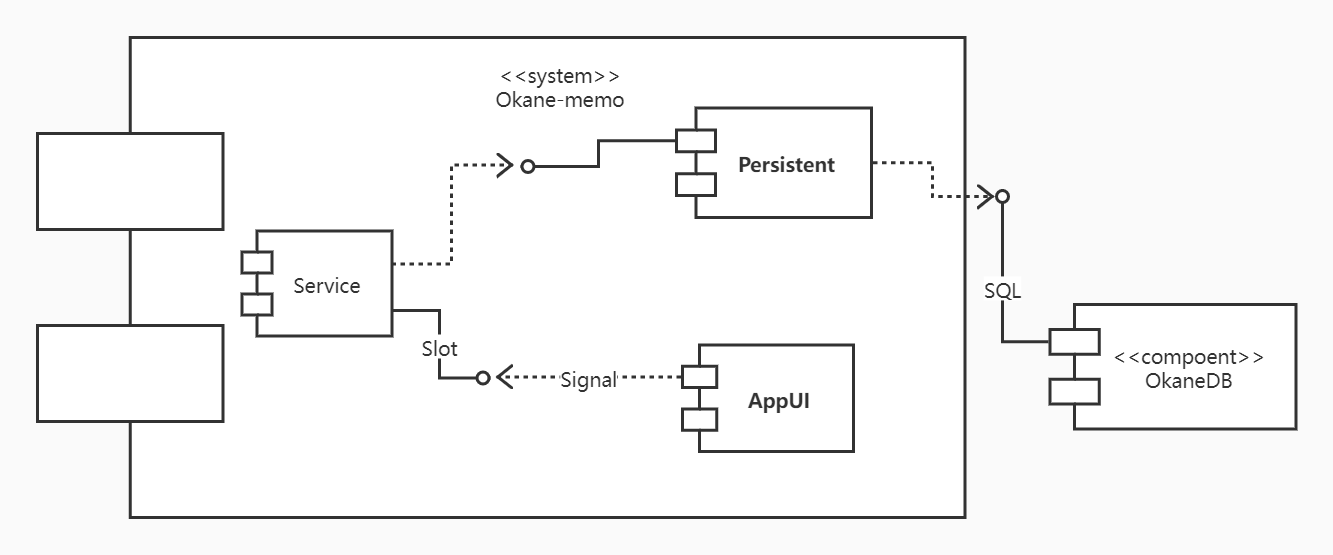


## 部署视图



## 实施视图

### 概述



### 层

* 1. UI层

此层包含表示用户看得到的应用程序屏幕的所有边界类。如MainActivity类，BookkeepingActivity类，SingleRecordActivity类

* 1. 业务层

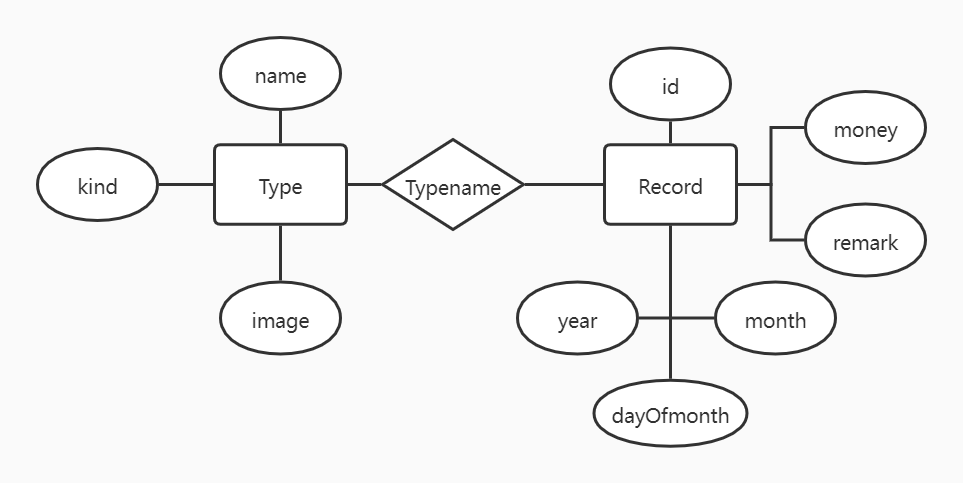
此层包括驱动应用程序行为的用例管理器的所有控制类。如RecordLinearAdapter类，TypeGridAdapter类

* 1. 数据持久层

此层负责与数据库进行交互。如DBManger类，OkaneDB类，RecordDao类

## 数据视图

### ER图



### 系统数据字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据名称 | 数据业务需求描述 | 业务数据类型 | 所属表格 | 对应的字段 | 字段类型 |
| **Type** | | | | | |
| **名称** | 唯一标识类型的字段 | 长度不超过20位的字符串，不能为空 | Type | name | char(60) |
| **类型** | 1表示为收入，-1表示为支出 | 1或者-1，不能为空 | Type | kind | int |
| **图片资源** | 标识类型的图片资源 | 图片的资源的整数编码 | Type | imageid | int |
| **Record** | | | | | |
| **记录ID** | 记录的ID号 | 2^31以下的整数 | Record | id | int |
| **金额** | 记录的金额大小 | 长度不超过 8位的浮点数，不允许为空 | Record | money | double |
| **备注** | 记录的补充信息 | 长度不超过200位的字符串，不能为空 | Record | remark | string |
| **年份** | 记录的年份 | 0~3000的整数 | Record | year | int |
| **月份** | 记录的月份 | 1~12的整数 | Record | month | int |
| **日期** | 记录在该月中的日期 | 1~31的整数 | Record | dayOfmonth | int |

## 质量

1. 可用性：用户可以正常记录信息，可以正确显示用户过往输入的数据
2. 性能：软件需要在 100ms 以内响应用户的动作，用户感觉不到等待
3. 可靠性：软件可以正常运行，存储大量数据时也不会发生崩溃
4. 可维护性：代码有良好可读性和可修改性，方便日后发布迭代版本