

# 软件设计文档

项目名称	基于 Android 的多功能日历的设计与实现
文档名称	软件设计文档
版本	v1.1
作者	张凯鑫
最后更新时间	2018-02-21

版本更新记录		
版本号	时间	更新内容
v1.0	2018-02-12	初稿
V1.1	2018-02-21	增加设计类图

# 目录

1. 引言 .....	3
1.1 编写目的 .....	3
1.2 读者对象 .....	3
1.3 软件应用概述 .....	3
1.4 文档概述 .....	3
2. 软件设计约束 .....	3
2.1 设计目标和原则 .....	3
2.2 设计约束 .....	4
3. 软件设计描述 .....	4
3.1 总体结构设计与模块划分 .....	4
3.2 设计类图 .....	4
3.3 接口设计 .....	5
3.4 数据库设计 .....	6

# 1. 引言

## 1.1 编写目的

编写本文档的目的是为了使开发更明确软件设计，便于在软件需求分析文档的基础上完成软件设计规定的各模块的具体需求。

## 1.2 读者对象

本文档的读者主要是应用开发人员、项目指导老师。

## 1.3 软件应用概述

### 1.3.1 软件名称

软件名称为“生活日历”。

### 1.3.2 软件功能概述

从整体上说，软件功能是围绕日历展开的，主要有添加日程备忘、写日记、设备周年纪念日、记录账单、查看天气、任务打卡等，将日常生活中与日期相关的功能细化，从而使得软件更具体、更有针对性。其中，除天气模块无数据库的使用，其他模块都涉及数据库的增、删、查、改功能。

### 1.3.3 软件性能要求

尽量减少卡顿现象的出现，尤其需要网络访问的部分应尽量避免网络迟缓给用户带来不好的用户体验，如果没有网络或者网络状况不足以加载数据则弹出一些用户友好的提示。

## 1.4 文档概述

本文档主要说明应用所选用的软件设计技术，并分析其整体架构、模块划分及接口设计、模块架构设计，并按此内容流程组织文档。

# 2. 软件设计约束

## 2.1 设计目标和原则

应用的设计应实现用户的需求，即实现需求规格说明书中所提及的全部功能。此外，应用具有良好的可扩充性，便于之后功能的添加与实现。在此基础上，应用设计应尽可能得清晰明确。

为实现上述目标，在设计软件过程中将会遵循软件设计的以下原则：

- 开闭原则：对组件功能的扩展是开放的，对原有代码的修改是封闭的。
- 依赖倒置原则：依赖于抽象，不依赖于实现，针对接口编程，不针对实现编程。

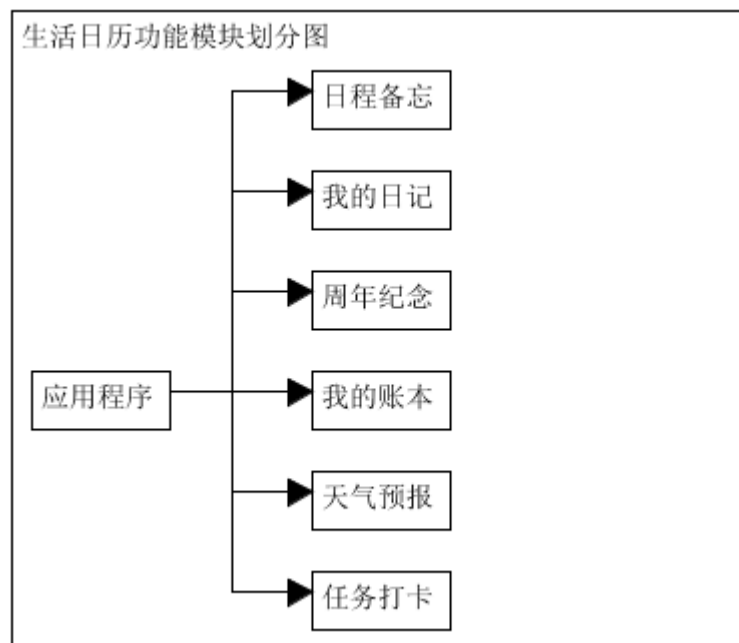
## 2.2 设计约束

- 硬件平台：android手机或模拟器
- 开发语言：Java
- 开发工具：Android Studio 2.3
- 云数据库：bmob 后端云

## 3. 软件设计描述

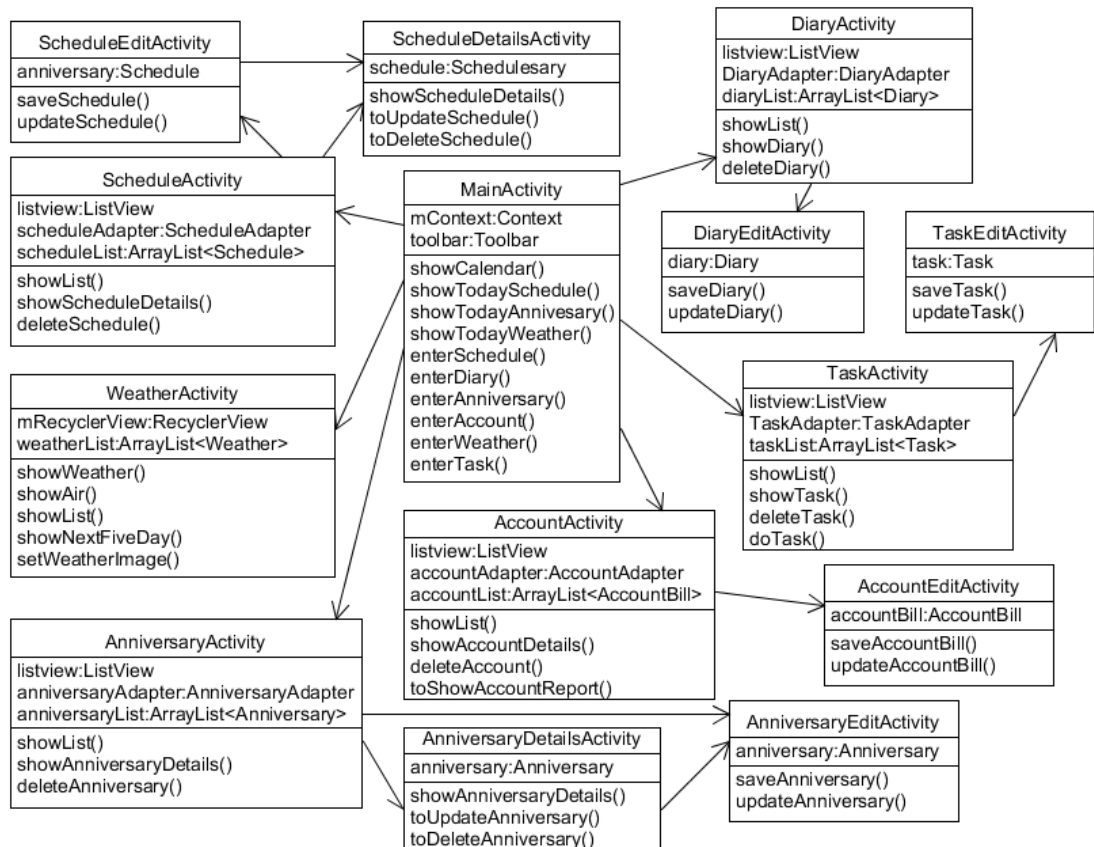
### 3.1 总体结构设计 with 模块划分

本应用分模块设计，每一个大功能为一个模块，根据应用的功能可划分如下图所示：



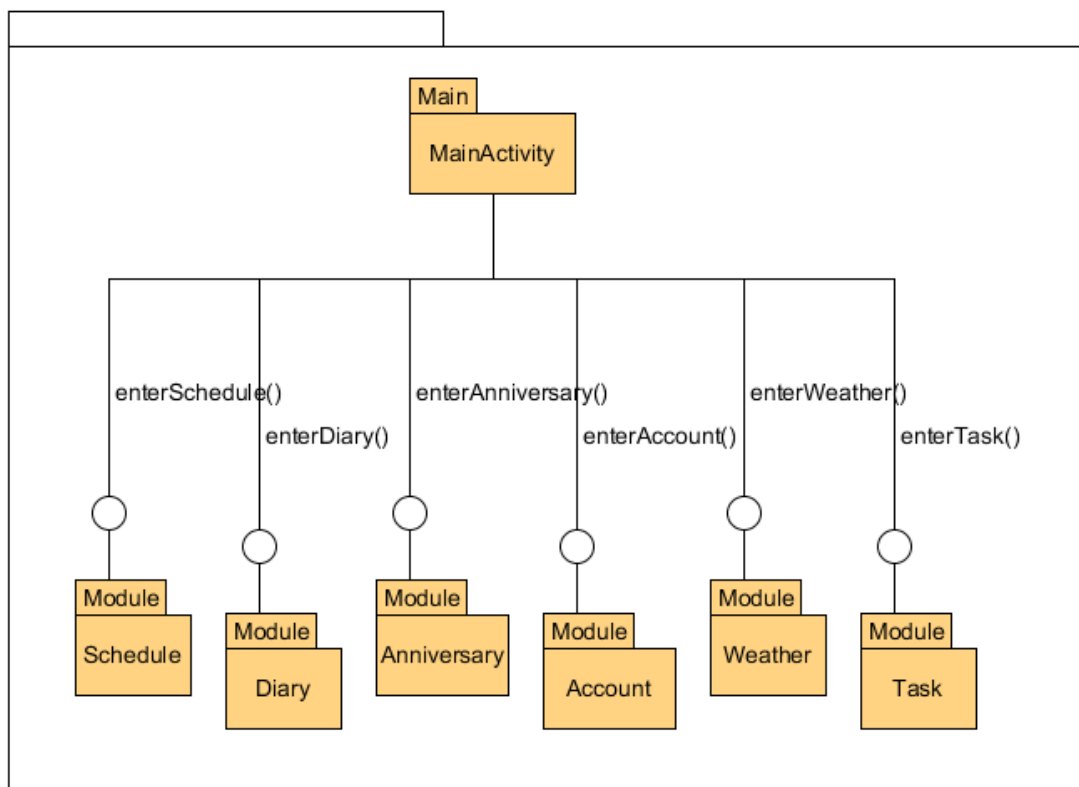
### 3.2 设计类图

初步分析项目需求，结合设计架构，本应用涉及到的 Activity 主要有以下几个： MainActivity 、 ScheduleActivity 、 ScheduleEditActivity 、 ScheduleDetailsActivity 、 DiaryActivity 、 DiaryEditActivity 、 AnniversaryActivity 、 AnniversaryEditActivity 、 AnniversaryDetailsActivity 、 AccountActivity 、 AccountEditActivity 、 TaskActivity 、 TaskEditActivity 、 WeatherActivity 等，Activity 的设计类图如下：

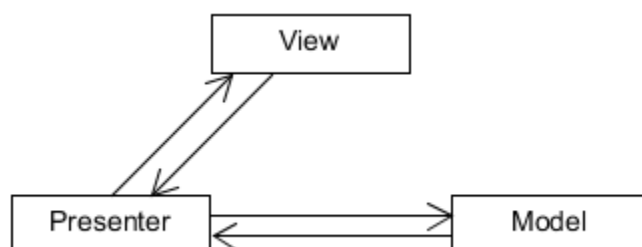


### 3.3 接口设计

本应用采用 android 抽屉式布局，拥有侧滑抽屉，侧滑出现功能菜单接口，可以跳转到模块，具体接口设计如下图所示：



各模块的实现使用MVP（Model-View-Presenter）架构设计模式， MVP是MVC架构的一种演变，分别表示数据层（Model）、视图层（View）、发布层（Presenter）。它将模型与视图完全分离，可以只修改视图而不影响模型，在MVP中，View不直接使用Model，它们之间的通信是通过Presenter来进行的，所有的交互都发生在Presenter内部。如下图：



此外，MVP 对于 android 应用开发有着很好的优势，它能实现模块分工，这能大大简化开发人员对代码的理解难度，遵循单一职责原则，将复杂的业务逻辑分解。它具有良好的易用性，而且维护成本低。

### 3.4 数据库设计

本应用使用到的数据库有两种，客户端使用的是 android 系统自带的 sqlite4 数据库，服务器端使用 bmob 后端云的数据库，两个数据库的表设计是一致的，便于上传和获取数据。数据库中的表设计如下：

表名：schedule_table（“日程备忘”表）			
名称	类型	约束条件	说明
id	String	无重复，不允许为空	主键
name	String	不允许为空,不超过 20	表示备忘的名称

		字符	
date	String	不允许为空	表示备忘的日期
notes	String	不超过 50 字符	表示备忘的备注内容，允许为空

表名: diary\_table ( “日记” 表)

名称	类型	约束条件	说明
id	String	无重复，不允许为空	主键
title	String	允许为空，不超过 20 字符	表示日记的标题
date	String	不允许为空	表示日记的日期
weather	String	不允许为空	表示日记的天气
address	String	不允许为空	表示日记的地点
content	String	不允许为空	表示日记的内容

表名: anniversary\_table ( “周年纪念” 表)

名称	类型	约束条件	说明
id	String	无重复，不允许为空	主键
name	String	不允许为空, 不超过 20 字符	表示纪念日的名称
date	String	不允许为空	表示纪念日的日期
notes	String	不超过 50 字符	表示纪念日的备注内容，允许为空

表名: account\_table ( “账单” 表)

名称	类型	约束条件	说明
id	String	无重复，不允许为空	主键
type	integer	不允许为空	表示账单的类型 ( “0” 表示 “支出”，“1” 表示 “收入” )
label	String	不允许为空	表示账单的标签 ( “0” 表示 “购物”，“1” 表示 “餐饮”，“2” 表示 “居住”，“3” 表示 “交通”，“4” 表示 “娱乐”，“5” 表示 “其他支出”，“6” 表示 “工资”，“7” 表示 “红包”，“8” 表示 “收益”，“9” 表示 “奖金”，“10” 表示 “报销”，“11” 表示 “其他收入” )
date	String	不允许为空	表示账单发生的日期
money	double	不允许为空	表示账单的金额
notes	String	不超过 20 字符	表示账单的备注内容，允许为空



表名: task_table ( “任务” 表)			
名称	类型	约束条件	说明
id	String	无重复, 不允许为空	主键
event	String	不允许为空, 不超过 20 字符	表示任务的名称
createdate	String	不允许为空	表示任务创建日期
notes	String	不超过 50 字符	表示任务的备注内容, 允许为空