

# Android 软件开发测试文档

## 1. 引言

### 1.1. 项目介绍

「倒数日」是一款倒数日应用，也是个纪念日应用和备忘录应用，可以帮助用户随时随地记录下自己的待办事项并且设置时间提醒功能，还可以将节日、纪念日记入 app 中，方便我们随时查看。

### 1.2. 职责范围

表 1 成员分工

成员	工作范围和工作内容
王衍飞（组长）	负责主界面、新增事件界面、取色算法开发
陈昱滔	负责主界面、剪贴板导入、扫码导入开发
谢凯	负责设计 bean 类、数据库
李林	负责设置界面与存储相关部件开发
张石一	负责桌面小部件开发
侯润宁	负责设计 dao 类、数据库

### 1.3. 开发条件

移动端开发使用 Kotlin 在 Android Studio 中编程，并且服务器可以部署于 Android 系统之上，以保证提供稳定可靠的服务。服务器系统硬件配置需满足服务器能够高效稳定地运行。

基于软件安全和保密性的考虑，系统的配置文件、数据存储文件等需要进行加密处理，采用国际通用的加密算法，防止意外泄露或恶意攻击。

基于系统可靠性的考虑，系统的数据存储文件应进行冗余备份，为了系统将来的可扩展性，系统硬件使用国际通用的硬件，不应使用具有针对性的硬件。整个系统也应尽量减少各模块间的调用，尽量做到松耦合。

### 1.4. 名词解释

表 2 名词解释

术语	定义
----	----

用户角色	用户角色是指按照一定参考体系划分的用户类型，是能够代表某种用户特征、便于统一，描述的众多用户个体的集合。
功能性需求	功能性需求规定开发人员必须在产品中实现的软件功能，用户利用这些功能来完成任务，满足业务需求。
非功能性需求	非功能性需求是指依一些条件判断系统运作情形或其特性，而不是针对系统特定行为的需求。
功能结构图	功能结构图是将系统的功能进行分解，按功能从属关系表示的图表。
用例图	用例图是指由参与者（Actor）、用例（Use Case）以及它们之间的关系构成的用于描述系统功能的静态视图。

表 3 缩写语

缩写语	定义
SDL	Specification and Description Language，规格和描述性语言，描述一个现实世界中特定事件的发生过程的时序关系以及步骤。
DFD	Data Flow Diagrams，数据流图从功能的角度对系统建模，追踪数据的处理有助于全面地理解系统，数据流图也可用于描述系统和外部系统之间的数据交换。
UML	Unified Modeling Language 统一建模语言或标准建模语言，为软件开发的所有阶段提供模型化和可视化支持，由需求分析到规格，再到构造和配置。

## 2. 软件概述

### 2.1. 项目概述

**产品名称：**倒数日

**产品目标用户：**在校学生、情侣、上班族等

**产品使用场景：**待办事项提醒、学习计划辅助、会议日程梳理、聚餐团建安排，以及为节假日与纪念日，恋爱累计天数提供时间记录与提醒

**产品设计背景：**一个特别的日期的背后，往往都有一段故事。于是，每一年，甚至每个月，都有那么些特别的日子值得铭记：奋战许久的高考日、与恋人初遇的日子、期待已久的欢聚日、至亲好友的生日、情人节、结婚纪念日……有时需要一种富含仪式感的方法，去随手记录生活中一些具有纪念意义的日期。

**产品目标：**营造一种生活的仪式感，传达「每一个日子都值得纪念」的生活理念。探索一种新的交互形式，使产品拥有更多的个性化设置，达到产品虽小但是灵活轻便的设计目标。

**市场前景：**市场上已经有一些类似功能的产品，如「365Days」「最美的时光」等，但是普遍存在设计结构混乱、用户个性化体验不佳、产品界面美观性不足等问题，我们针对如上缺陷，制订了集设计感、交互性、个性化为一体的轻量级产品，深入满足目标客户在工作中的秩序感与生活中的仪式感的双重要求。同时，通过收集用户特征数据，后期不断迭代，完善产品功能，丰富用户体验。通过满足不同群体需求，下沉市场空间很大，市场发展前景良好。

2. 2. 功能概述

表 4 产品功能概述

角色	功能
用户	1. 添加事项 2. 数据备份 3. 事件分享 4. 排序

**提醒功能**

业务需求：根据用户的设置，按时提醒用户倒数日/提醒日距目标日的时间

业务目标：提醒用户倒数日/提醒日距目标日的时间

业务目的：及时提醒用户，提高用户的使用频率

衡量指标：日活跃用户数

用户行为：通过通知条打开产品

**添加事项**

业务需求：用户通过添加倒数日/纪念日来记录目标事件

业务目标：记录用户的重要目标事件

业务目的：通过记录目标事件营造一种仪式感

衡量指标：目标事件生成数量

用户行为：通过添加按钮添加目标事件

**备份功能：**

业务需求：用户注册以后可以在云端保存数据，防止数据丢失

业务目标：提高注册转化率

业务目的：提高用户的留存率

衡量指标：注册用户数量

用户行为：完成备份流程

分类功能：

业务需求：用户可以对目标日进行分类，并进行筛选

业务目标：引导用户对目标日进行分类管理

业务目的：提高用户对目标日进行管理的效率

衡量指标：用户创建分类的数量

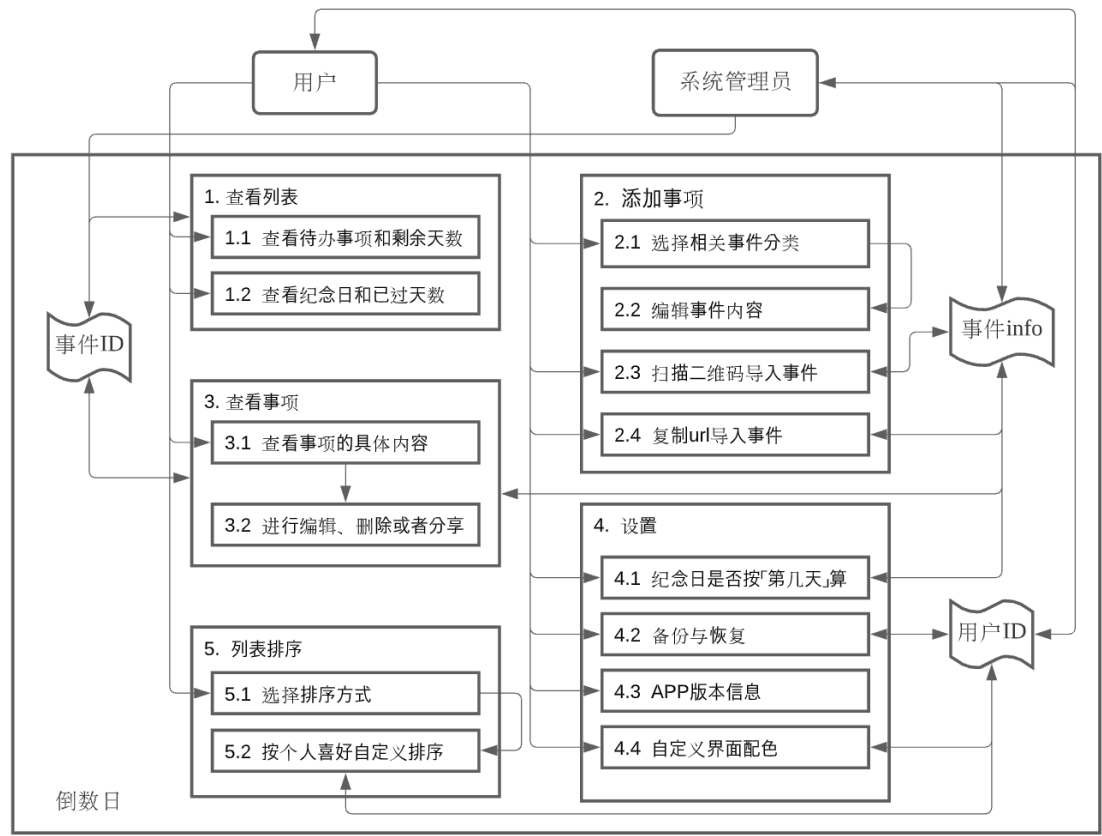


图1 产品第二层数据流图

2. 3. APP 特色

- ✧ 在原有倒数日的功能基础上，增设事件种类为「纪念日」「生日」「农历生日」「倒数日」，分别对应不同的日期计算方式，用户也可以自定义事件分类与计算方式。
- ✧ 提醒与备忘录功能二合一

- ✧ 可以复制事件 URL 或扫描二维码一键导入已有的事件信息，方便用户之间分享事件，不需要重新设置相同的事件信息，操作灵活快捷。
- ✧ 可以自定义 RGB 的界面配色，按照个人喜好定制个性化的 APP 界面。
- ✧ 可以在排序中选择「按添加时间从旧到新」「按添加时间从新到旧」「自定义排序」「按天数从少到多」「按天数从多到少」。用户也可以滑动事件块自定义排序，以个性化、私人化、微小化的设置，与市面上千篇一律的纪念日产品区别开来。
- ✧ 具有备份与恢复功能，用户可以在不同设备上导入数据，不用担心数据丢失。

### 3. 需求分析

#### 3.1. 适用人群

本产品面向 15 岁到 45 岁的人群，包括在校学生、上班族、情侣等，提供待办事项提醒、学习计划辅助、会议日程梳理、聚餐团建安排，以及为节假日与纪念日，恋爱累计天数提供时间记录与提醒功能。此外针对不同用户的多样需求提供个性化、私人化、微小化的设置选项，为客户群体量身定制属于自己的时间刻画，平衡学业、工作、人际关系往来的多项社会角色，让生活丰富多彩又不失秩序井然。

#### 3.2. 运行环境

- 硬件要求："armeabi", "armeabi-v7a", "arm64-v8a", "x86"处理器的 Android 设备
- 最低支持的系统版本：Android 7.0 （minSdkVersion 24）
- 推荐的系统版本：Android 10 （targetSdkVersion 29）

### 4. 业务设计

#### 4.1. 业务流程

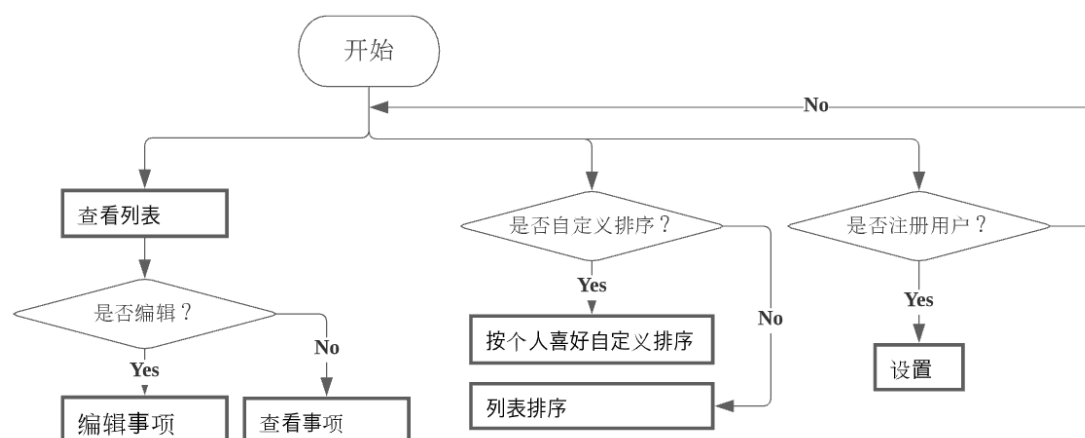


图 2 业务流程图

#### 4.2. 用户用例图 (User Case)

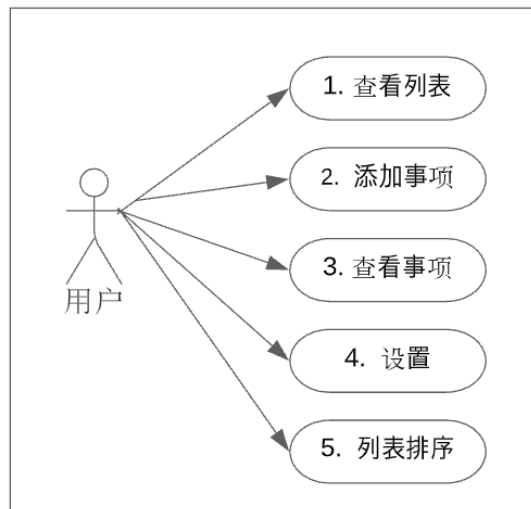


图 3 用户用例图

APP 用户主要参与的用例有查看事件列表、添加事项、查看事项内容、设置与事件列表排序等功能。

4.3. 用户界面



事件列表



添加事件



编辑事件



查看事件



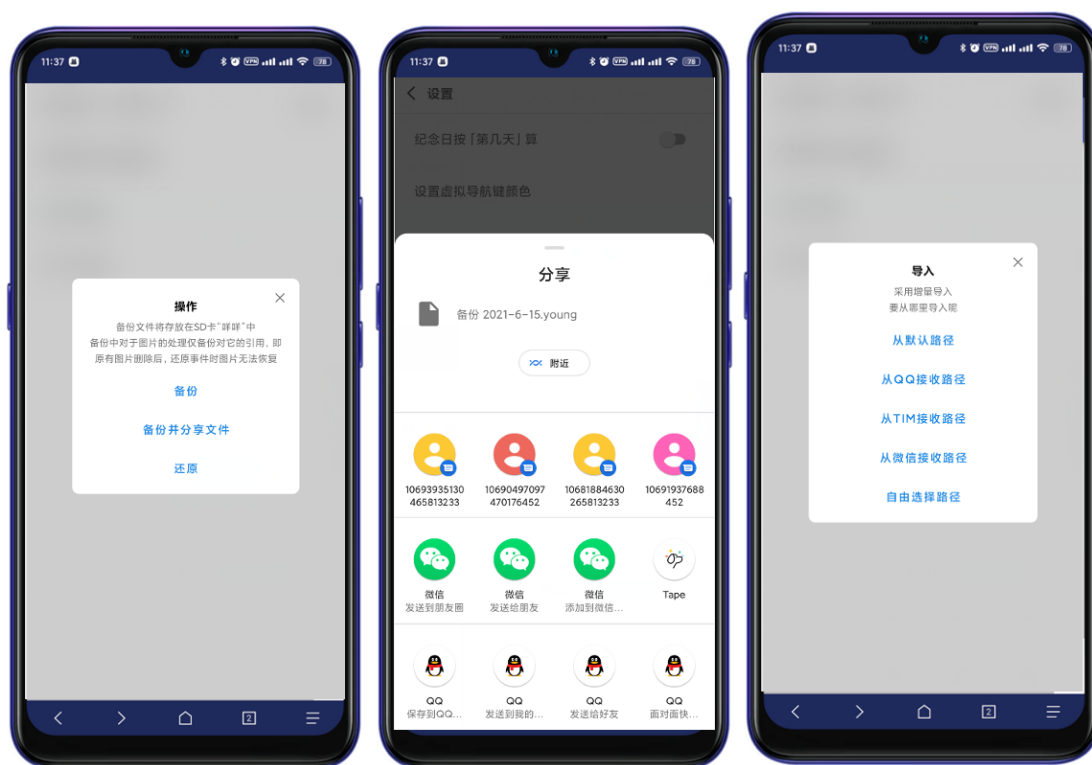
分享事件



排序事件



分享功能保存为图片样例



备份与恢复



4.4. 角色操作流程

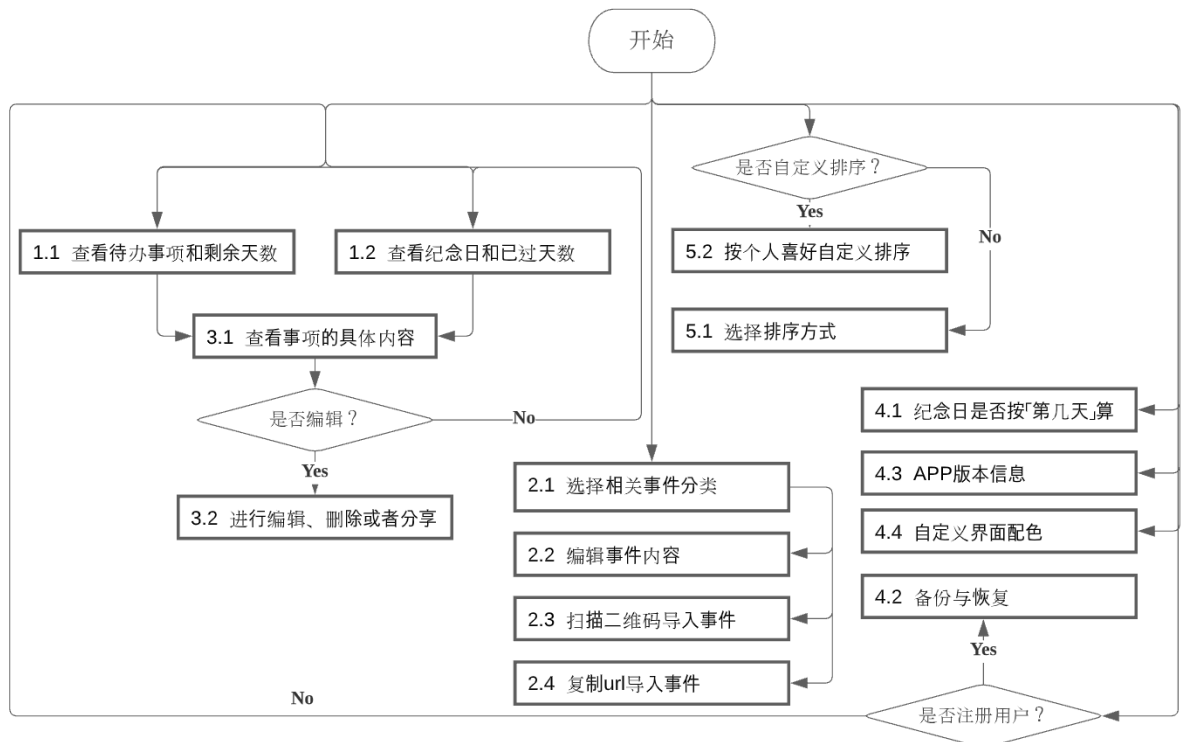


图 4 用户操作流程图

5. 功能设计

5.1. 功能划分

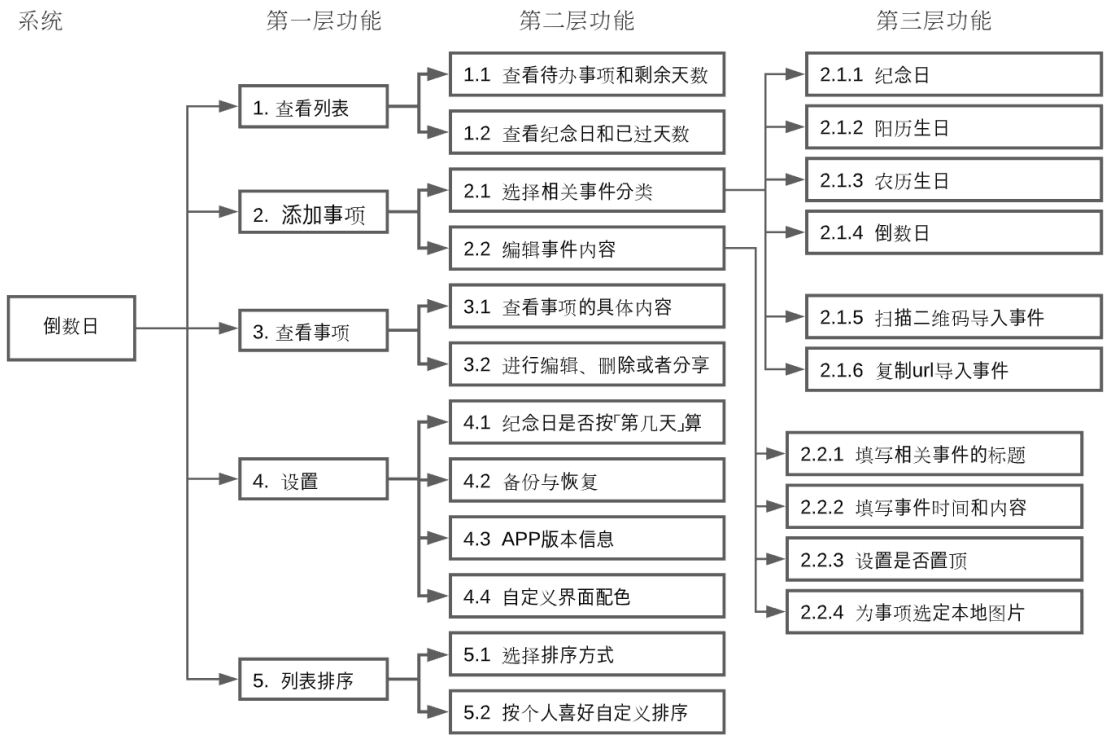


图 5 功能模块结构图

## 5.2. 功能详述

### 5.2.1 查看列表

介绍	用户登陆 APP 主界面后可以查看所有事件的列表
输入	用户的 ID 信息
过程	在主界面中可以查看已经设定好的：待办事项日期和剩余天数、纪念日的日期和已过天数  主界面顶栏有设置、添加和设定排序的菜单，用户可以按照自己的爱好和需要设置列表的排序、虚拟导航按键的色彩等等
输出	所有事件的列表与用户的个人配置
错误处理	系统显示错误信息，用户需要联系系统管理员，重新连接数据库

### 5.2.2 添加事项

介绍	用户点击「添加事件」，可以编辑新的事件并加入事件列表
输入	事件的分类与内容，或者从已有事件的 URL 或二维码一次性导入
过程	用户点击添加事件后，可以选择「新建事件」「扫码导入事件」「从 URL 导入事件」。新建事件时，可以选择事件种类为「纪念日」「生日」「农历生日」「倒数日」，分别对应不同的日期计算方式。可以填写相关事件的标题、事件时间和内容、是否设为置顶，以及为事项选定本地图片
输出	事件列表中的新事件
错误处理	系统提示 URL 不存在或二维码信息错误，用户需要重新导入正确的事件 URL 与二维码

### 5.2.3 查看事项

介绍	用户添加完需要保存的记录后，可以在主页查看事项
输入	用户点击的事件 ID
过程	用户点击「查看事项」后，可以点击任一事项，查看事项的具体内容。可以选择对该事项进行编辑、删除或者分享。
输出	事件内容与详细信息
错误处理	系统提示事件不存在，用户刷新页面，更新已被删除的事件和当前存在的事件列表。

5.2.4 设置

介绍	用户点击「设置」后，可以对事件进行设置，按个人喜好自定义系统功能，查看版本信息
输入	用户设置的属性
过程	用户可以在设置中确认是否纪念日按"第几天"算；APP 提供备份与恢复的功能；「关于」中供用户查看 app 版本信息；按照个人喜好自定义系统配色。
输出	系统信息与更新配置
错误处理	系统更新配置失败，用户需要连接网络重新尝试

5.2.5 列表排序

介绍	用户按照个人喜好选择排序方式，也可以自定义排序
输入	用户设置的排序方式
过程	用户可以在排序中选择「按添加时间从旧到新」「按添加时间从新到旧」「自定义排序」「按天数从少到多」「是否按天数从多到少」。用户也可以滑动事件块自定义排序。
输出	系统更新事件排序
错误处理	系统更新自定义排序失败，用户需要重新设置排序方式。

5.3. 功能模块交互流程图

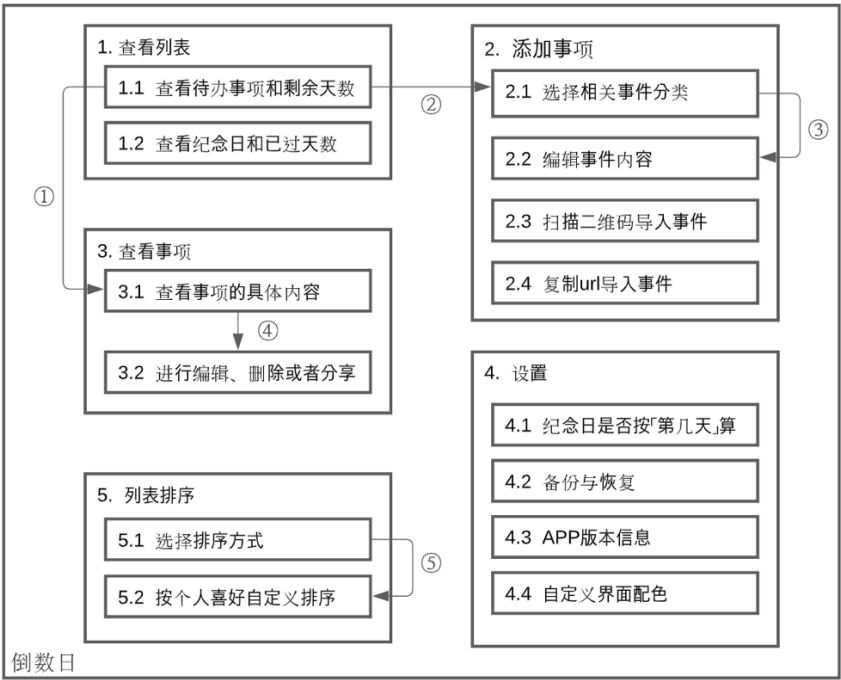


图 6 功能模块交互图

### 模块交互①：查看事件

当用户登陆主界面，查看列表所有待办事项与剩余天数时，点击单个事件组件，从功能 1.1 跳转到 3.1，转到单个事件内容页面，可以查看事项具体内容。

### 模块交互②：添加事件

当用户点击添加事件按钮时，从功能 1.1 跳转到 2.1，从登陆时的主界面转到添加事件的分类页面，可以选择事件分类，或从 URL 与二维码一键导入事件。

### 模块交互③：添加描述

当用户在事件分类中没有选择「扫描二维码导入事件」「复制 URL 导入事件」，而是在系统自己设定的事件分类「纪念日」「倒数日」「农历生日」「阴历生日」或「自定义分类」中任选一项时，从功能 2.1 跳转到 2.2，用户跳转到事件描述页面，添加事件时间与内容等基本信息。

### 模块交互④：编辑事件

当用户查看事项具体内容时，从功能 3.1 跳转到 3.2，可以选择对事项进行进一步操作，如编辑内容、删除事项，或者分享给好友。

### 模块交互⑤：自定义排序

当用户选择排序方式，选择「自定义排序方式」时，从功能 5.1 跳转到 5.2，用户跳转到自定义排序方式页面，设置个性化的排序计算方式。

## 6. 接口设计

### 6.1. 硬件接口

无

### 6.2. 软件接口

无

### 6.3. 通信接口

#### 6.3.1 扫 QR 码分享事件

使用开源库 zxing 生成二维码。二维码内容的数据为该事件所包含的内容等信息，数据结构为使用::分隔的六元组，格式为：事件名::类型::年::月::日::备注。如 test::0::2021::5::16::

#### 6.3.2 从网络导入事件（通过 URL）

使用开源库 okhttp 处理 HTTP/HTTPS 请求。以 GET 形式对 URL 进行请求，返回的 BODY 部分为数据，格式为::分隔的七元组，分别为：事件名::类型::年::月::日::备注::图片 URL。当

有多个事件时使用##分隔，如：test::0::2021::5::16:::##test2::0::2021::4::13:::

### 6.3.3 备份与恢复

使用 Kotlin 序列化库以 Json 的形式序列化数据保存为文件。

## 7. 数据描述

### 7.1. 数据管理要求

需要管理的数据均使用关系型数据库存储，规模通常在几十到上百条记录，相对较小。

### 7.2. 数据库描述

事件数据库

键值	类型	备注
id	Int	主键
content	String	
year	Int	
month	Int	
day	Int	
type	Int	
path	String	
isFav	Bool	
msg	String	
sortNum	Int	

桌面小组件数据库

键值	类型	备注
id	Int	主键
eventId	Int	外键
withPic	Bool	
weight	Int	
bgColor	Int	
textColor	Int	
textHorizontal	Bool	
contentSize	Int	
daySize	Int	
msgSize	Int	

### 7.3. 输入和输出数据

- year、month、day：任意合法的日期
- content、msg：字符串类型，没有长度限制

- isFav : Bool 类型, 真或假
- type : 整形, 但只能为 1 ~ 4 其中之一

#### 7.4. 数据字典

无

#### 7.5. 数据采集

- 用户输入数据: 用户通过 add\_event\_activity 添加新事件
- 扫码导入数据: 用户通过扫描二维码添加新事件
- 剪贴板导入数据: 用户通过剪贴板导入并添加新事件
- URL 导入数据: 用户通过 HTTP 协议从互联网上获取并添加新事件

### 8. 性能要求

#### 8.1. 精度

纪念日事件精确到天

类型目前支持 4 种: 纪念日、生日, 农历生日, 倒数日

#### 8.2. 实时性

新建纪念日事件响应时间: 0.2s

更新纪念日事件响应时间: 0.2s

#### 8.3. 灵活性

兼容安卓 9 以及鸿蒙 2.0 系统

兼容最低安卓 API 21 最高 API29

#### 8.4. 鲁棒性

系统能较长时间下稳定运行, 同时该系统需要具备一定的故障恢复能力, 即有一定的容错能力。当用户的操作不当引起某些故障时, 或者是由于操作系统或者网络发生故障时, 系统需要具备一定的故障恢复能力。

选择数据库产品时, 要考虑一定的数据负载能力。由于在处理用户配置、事件内容等信息时, 系统需要做大量的数据统计和处理, 因此要具备相应的数据负载能力

系统响应时间考虑到同时进行操作的用户人数, 倒数日系统应当能最低支持 10000 位用户, 期望正常支持 20000 位用户, 在支持 30000 名用户时系统仍能正常使用, 支持高并发操作。

## 9. 测试报告

### 9.1. 测试环境描述

软件平均响应速度：250ms

### 9.2. 功能模块单元测试

我们对 Utils 目录下的一些辅助函数进行了单元功能测试，测试覆盖 100%，通过 100%。

### 9.3. 角色业务流程测试

使用 espresso 进行 UI 自动化测试，测试结果如下：

- 打开 MainActivity，成功；
- 从 MainActivity 跳转到 SettingsActivity，成功；
- 从 MainActivity 打开 DetailFragment，成功；
- 从 MainActivity 跳转到 ScanQRActivity，成功；

### 9.4. 压力（性能）测试

使用 monkey 对 App 进行压力测试。

App 后台常驻内存使用：30M

在 CPU 占用超过 98%时或者内存不足的情况下，应用出现明显的卡顿。表现为点击事件响应缓慢卡顿，打开 APP 缓慢。

### 9.5. 故障描述

- 更新纪念日界面布局不符合预期，通过调整布局修复。
- 纪念日类型标签堆叠，与预期不符。通过调整布局以及展示层级修复。
- 从剪切板创建新纪念日失败
- 从剪切版将信息导入数据库时，规则字符串解析错误。需要使用对应格式的新建纪念日规则来导入。

### 9.6. 整体评估

经过测试，我们的安卓应用已经基本可用。在已经存在的功能上，没有发现有任何重大故障。

但应用仍有较大提升空间，有许多功能（如循环事件）尚未实现。

## 10. 参考文档

[1] 王安生.《软件工程化》[M]. 北京：清华大学出版社，2014.

[2] Hardy, B., & Phillips, B. (2013). *Android programming: The big nerd ranch guide*. Addison-Wesley Professional.