

LetsToDo 系统设计说明书

LetsToDo 系统设计说明书

1. 引言

- 1.1 编写目的
- 1.2 读者对象
- 1.3 术语与缩略语解释
- 1.4 参考资料

2. 总体设计

- 2.1 需求规定
- 2.2 运行环境
- 2.3 软件功能结构设计
 - 2.3.1 层次图
 - 2.3.2 类图

3. 系统模块设计

- 3.1 事件管理模块
 - 3.3.1 模块功能描述以及划分
 - 3.3.2 接口设计
- 3.2 日期管理模块
 - 3.2.1 模块功能描述以及划分
- 3.3 番茄时钟模块
 - 3.3.1
- 3.4 个人中心模块

4. 系统安全和权限设计

- 4.1 系统安全
 - 4.1.1 开发过程遵循原则
 - 4.1.2 HTTPS 证书
- 4.2 权限设计

1. 引言

1.1 编写目的

主要是说明“LetsToDo”小程序的系统结构以及内部设计，提供详细的系统结构设计、功能模型和接口设计描述。用于指导该项目的开发，为未来的系统维护和功能扩展做参考资料。

1.2 读者对象

- 产品经理：在项目这个生命周期中不断完善此文档，以此文档为基准，进行管理、分工和验收。
- 设计师：根据本文档的功能模块设计进行相关的具体工程结构设计、用户界面设计等。
- 开发人员：通过本文档属性数据库设计和功能要求，并以此来进行具体的编码。
- 测试人员：根据本文档对所有功能和界面进行充分的测试，以验收标准来作为测试的标准。
- 用户：了解具体运行方法和所具有的功能。
- 其他人员：根据本文档了解“LetsToDo”的系统设计。

1.3 术语与缩略语解释

| 术语、缩略语 | 说明 |
|----------|---|
| LetsToDo | 我们所设计的小程序名称，即 Let's to do，去除了空格以及分隔符之后所得。意为：让我们行动起来。作为小程序的名称，能够让用户快速了解到我们的小程序是一款“todolist”类型的小程序。 |

1.4 参考资料

| 标识符 | 文献名称 | 作者 | 出版单位（或归属单位） | 出版日期 |
|-----|--------------|------------|-------------------------|------------|
| a | LetsToDo原型设计 | Double-Ten | 福州大学数计学院 2018 级软件工程 Z 班 | 2020-11-02 |

2. 总体设计

2.1 需求规定

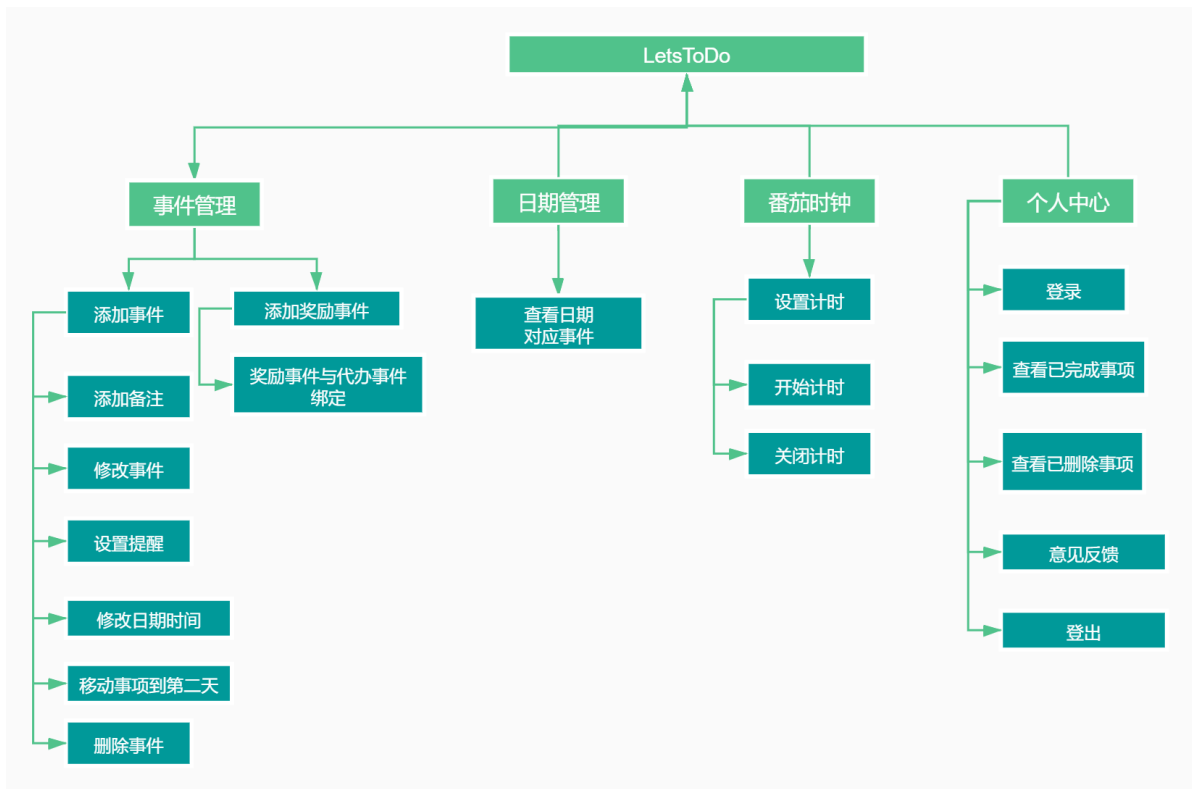
| 功能 | 输入 | 处理 | 输出 |
|---------|------|-------|--------|
| 用户登入 | 用户信息 | 查询数据库 | 用户信息 |
| 创建事件 | 事件信息 | 添加数据库 | 事件信息 |
| 完成事件 | 事件信息 | 修改数据库 | 事件信息 |
| 删除事件 | 事件信息 | 修改数据库 | |
| 过期未完成事件 | 事件信息 | 修改数据库 | |
| 设置番茄钟 | 时间信息 | | 时间倒数 |
| 日历同步 | 事件信息 | 添加数据库 | 事件信息 |
| 提醒功能 | 时间信息 | 添加数据库 | 提醒信息 |
| 查看已完成事件 | 事件信息 | 查询数据库 | 完成事件列表 |
| 查询已删除 | 事件信息 | 查询数据库 | 删除事件列表 |

2.2 运行环境

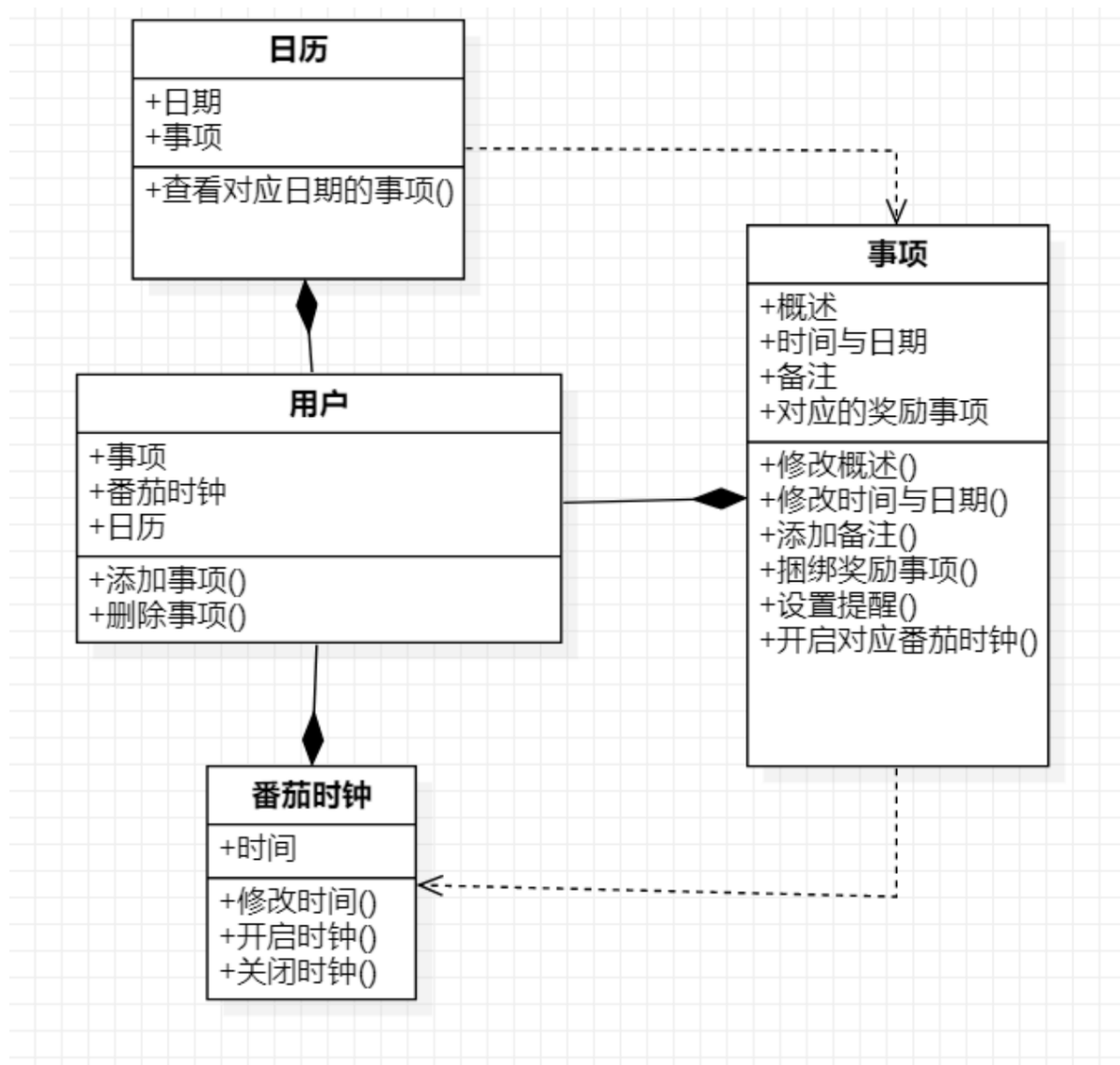
- 1. 运行安卓系统的移动设备
- 2. 运行 IOS、ipadOS 的移动设备
- 3. 运行 windows/macOS 的电脑

2.3 软件功能结构设计

2.3.1 层次图



2.3.2 类图



3. 系统模块设计

3.1 事件管理模块

该模块主要为用户提供对事件的管理，包括但不限于添加删除事件、设置提醒日期时间等。

3.3.1 模块功能描述以及划分

普通事件管理

| 功能 | 功能描述 |
|----------|----------------------------------|
| 添加事件 | 输入事件信息，添加进代办事件列表 |
| 删除事件 | 从代办事件列表中移出事件，并加入已删除事件列表 |
| 修改事件 | 修改事件的信息，备注，提醒日期时间等 |
| 设置提醒 | 设置对应的日期时间，在对应时间通过微信通知服务提醒用户 |
| 添加备注 | 编辑事件更为详细的描述 |
| 完成事件 | 从代办列表中删除，并加入已完成事件列表 |
| 过期未完成的事件 | 前一天未完成的事件，在第二天提醒用户是否添加成为第二天的代办事件 |

奖励事件管理，基本功能与普通事件相同

| 功能 | 功能描述 |
|-------------|---|
| 奖励事件与普通事件绑定 | 在建立普通事件时，用户可选择是否同步创建奖励事件。当完成普通事件后，奖励事件将自动加入代办列表 |

3.3.2 接口设计

| 接口类型 | 接口函数 |
|--|---|
| 登录 | wx.login(Object object) |
| 获取当前帐号信息。 | Object wx.getAccountInfoSync() |
| 获取用户信息。（调用前需要 用户授权 scope.userInfo） | wx.getUserInfo(Object object) |
| 用户信息 | UserInfo |
| 调起客户端小程序设置界面，返回用户设置的操作结果 | wx.openSetting(Object object) |
| 订阅消息设置 | SubscriptionsSetting |
| 调起客户端小程序订阅消息界面，返回用户订阅消息的操作结果。 | wx.requestSubscribeMessage(Object object) |

3.2 日期管理模块

该模块提供一个日历页面，以及对应日期的事件

3.2.1 模块功能描述以及划分

| 功能 | 功能描述 |
|----------|------------------------------|
| 查看日期对应事件 | 用户点击日历上的某一天，下方既可以出现对应日期设定的事件 |

3.3 番茄时钟模块

该模块为用户提供一个番茄时钟，用户可通过设置倒计时来更加专注的完成待办事项。

3.3.1

| 功能 | 功能描述 |
|------|-----------------------------|
| 设置计时 | 用户可自定义在 0~59:59 中任选一个时间开始计时 |
| 开始计时 | 在用户设定完时间之后，番茄钟开始倒计时 |
| 计时结束 | 在计时完成之后，提醒用户 |

3.4 个人中心模块

该模块为用户可给用户展示已完成，已删除的事件，以及对一些基本个人信息的管理。

| 功能 | 功能描述 |
|---------|--------------------------------|
| 登录 | 调用微信小程序接口，获取用户的个人信息 |
| 登出 | 退出小程序 |
| 查看已完成事项 | 返回已经完成的事项列表，展示给用户，包括事件名称，完成的时间 |
| 查看已删除事项 | 返回已经删除的事项列表，展示给用户，包括事件名称，删除的世界 |

4. 系统安全和权限设计

4.1 系统安全

4.1.1 开发过程遵循原则

- 互不信任原则，不要信任用户提交的数据，包括第三方系统提供的数据，必要的数据库校验必须放在后台校验。
- 最小权限原则，代码、模块等只拥有可以完成任务的最小权限，不赋予不必要的权限。
- 小程序代码（不包括云函数代码）跟传统 Web 应用的前端代码类似，可被外部获取及进行反混淆，重要业务逻辑应放在后台代码或云函数中进行。
- 后台接口调用以及云函数调用，必须进行有效的身份鉴权。

4.1.2 HTTPS 证书

必须使用 HTTPS/WSS 发起网络请求。 请求时系统会对服务器域名使用的 HTTPS 证书进行校验，如果校验失败，则请求不能成功发起。

4.2 权限设计

因此项目是一个“Todolist”类型的小程序，更多是用户自己使用，所以权限设计也比较简单。对用户权限分为登录用户和非登录用户两种权限：

- 登录用户可以使用小程序的全部功能。
- 非登录用户不能使用查看已完成和已删除事项的功能，且不保存数据，退出小程序则数据清空。

管理员权限：定期删除长期未登录的用户数据，以六个月为例。