|  |  |
| --- | --- |
| **案卷号** |  |
| **日期** |  |

翱翔清单

**详细设计说明书**

作 者：刘黎可、傅钰、徐薏茜、李振国、

于李源、贾聪毅、韩紫琪、张书畅

完成日期： 2024.01.12

签 收 人： 刘黎可 、 贾聪毅 、 张书畅

签收日期： 2024.01.14

修改情况记录：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本号 | 修改批准人 | 修改人 | 安装日期 | 签收人 |
| 1.0.0 | 刘黎可 | 韩紫琪 | 2024.1.2 | 刘黎可 |
| 1.0.1 | 张书畅 | 刘黎可 | 2024.1.9 | 张书畅 |
|  |  |  |  |  |

**目录**

[详细设计说明书 1](#_Tocjq6x4i)

[目录 1](#_Toccm4j46)

[1 引言 4](#_Tocpx9hda)

[1.1 编写目的 4](#_Tocsamedf)

[1.2 背景 4](#_Toc24ualb)

[1.3 定义(未完成) 4](#_Tocdxxvqo)

[1.4 参考资料 5](#_Tochh5xir)

[2 程序系统的结构 5](#_Tocx0bdpn)

[3 客户端前端设计说明 6](#_Toco0dwoh)

[3.1 程序描述 6](#_Tocd0fjg1)

[3.2 功能 8](#_Tocrxo4mc)

[3.3 性能 10](#_Tocronqev)

[3.4 输入项 10](#_Toczvsffj)

[3.5 输出项 11](#_Tocecrsuo)

[3.6 算法 11](#_Tocada87j)

[3.7 流程逻辑 11](#_Toc6f2n60)

[3.8 接口 14](#_Tocnlb0z9)

[3.9 存储分配 14](#_Tocgs7a3a)

[3.10 注释设计 15](#_Toc7271vp)

[3.11 限制条件 15](#_Toca3ot60)

[3.12 测试计划 15](#_Tocxtrszg)

[3.13 尚未解决的问题 15](#_Tocinvqcz)

[4 客户端后端设计说明 15](#_Tocx4vcfy)

[4.1 程序描述 16](#_Toc5v9vco)

[4.2 功能 16](#_Toc2j27ar)

[4.3 性能 20](#_Tocis6rxp)

[4.4 输入项 21](#_Tocfgpdbm)

[4.5 输出项 21](#_Toc8s9vu5)

[4.6 算法 22](#_Tocfw2c36)

[4.7 流程逻辑 22](#_Toclehvrk)

[4.8 接口 22](#_Tocozdkff)

[1.接口1 22](#_Tocmj006i)

[2.接口2 23](#_Toc0uuupy)

[4.9 存储分配 24](#_Tocst0w75)

[4.10 注释设计 25](#_Tocqh5a1i)

[4.11 限制条件 25](#_Tocfi2fee)

[4.12 测试计划 25](#_Tocopdel1)

[4.13 尚未解决的问题 26](#_Toclemg1a)

[5 服务端设计说明 26](#_Tocw68n0s)

[5.1 程序描述 26](#_Tocbvdz33)

[5.2 功能 26](#_Tocdh46mx)

[5.3 性能 27](#_Toczs2zd2)

[5.4 输入项 27](#_Tocz918r6)

[5.5 输出项 29](#_Toc9ym75z)

[5.6 流程逻辑 31](#_Toc56jlde)

[5.7 接口 32](#_Tocy30jo1)

[1.接口1 32](#_Toc7nlztq)

[2.接口2 33](#_Tocpz4a8n)

[3.接口3 33](#_Tocap0eyl)

[5.8 存储分配 34](#_Tocbn2gcv)

[5.9 注释设计 35](#_Toc6men8g)

[5.10 限制条件 35](#_Tocvwksbw)

[5.11 测试计划 35](#_Tocfomk2h)

[5.12 尚未解决的问题 36](#_Tocg0ndl1)

## 1 引言

### 1.1 编写目的

本详细设计说明书旨在明确“翱翔清单”系统开发过程中各功能模块的主要处理逻辑和流程，供开发人员参考使用。

### 1.2 背景

a. 软件系统名称：翱翔清单

b. 项目相关信息

|  |  |  |
| --- | --- | --- |
| 项目名称 | 对象 | 备注 |
| 任务提出者 | 刘黎可 |  |
| 开发者 | 本项目小组8位成员 |  |
| 用户 | 本课程相关人员 |  |
| 运行软件的单位 | 用户（客户端）及本小组（服务器） |  |

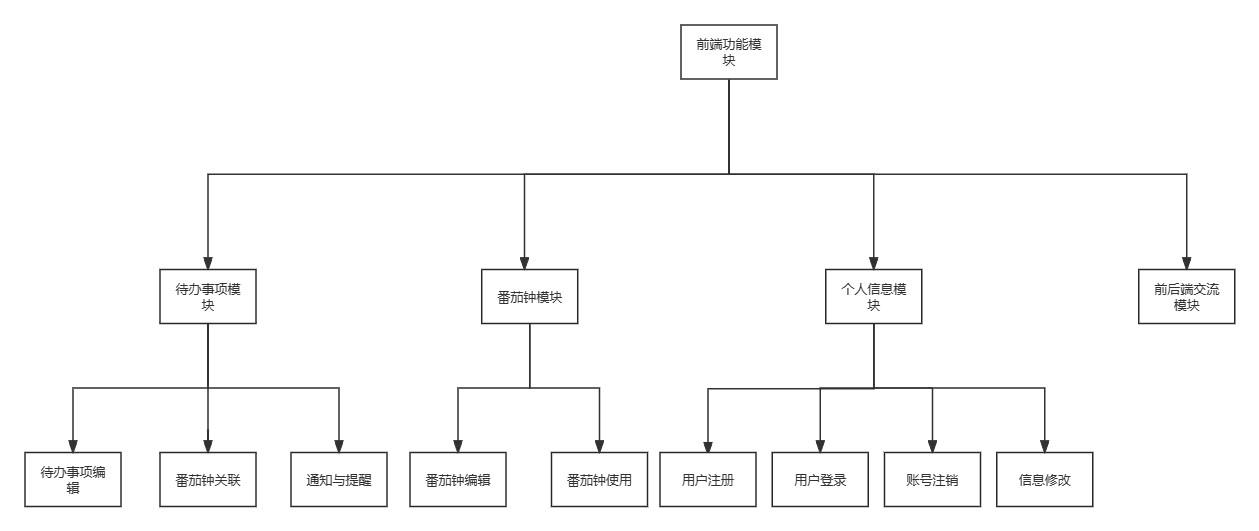
### 1.3 定义

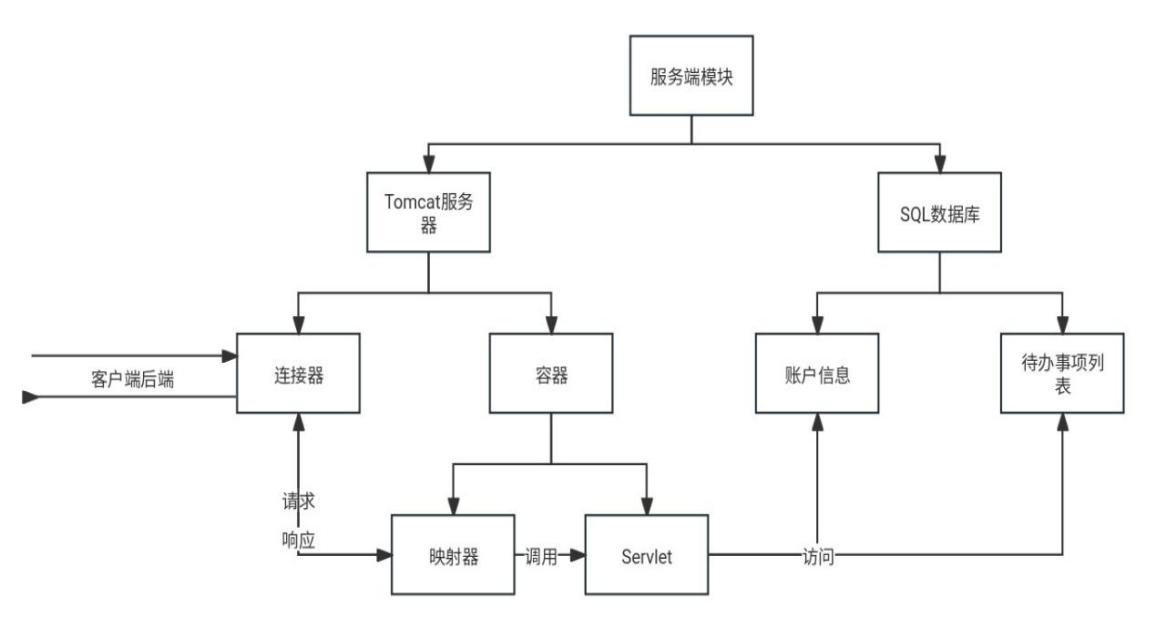
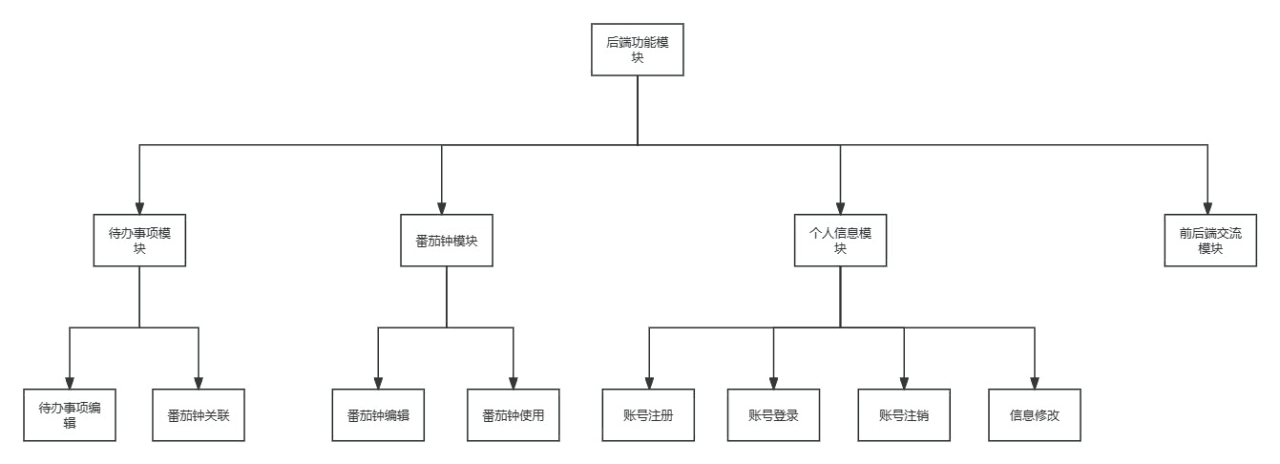
|  |  |
| --- | --- |
| 账号 | 用于标识和认证用户身份的数字实体，存储了用户在翱翔清单中的相关信息。 |
| 用户名 | 账号的元素之一，用于唯一标识系统用户的身份字符串。 |
| 密码 | 由用户设置的用于验证用户身份的保密数据字符串 |
| 待办事项 | 具有特定描述和期望结果的、等待用户处理的尚未完成的任务。 |
| 待办事项标题 | 对待办事项的高度概括性描述，包含事项的核心内容和目的。 |
| 待办事项描述 | 由用户撰写的、针对待办事项详细内容的文本性说明。 |
| 待办开始时间 | 待办事项预期执行的开始时间点 |
| 待办结束时间 | 待办事项预期完成的截止时间点 |
| 任务重要程度 | 待办事项的重要级别，表示该待办相对于其他任务的重要性 |
| 任务紧急程度 | 待办事项的紧急程度，表示该待办相对于其他任务的紧急性 |
| 番茄钟 | 用户设置的完成该待办事项的任务时间。 |

### 1.4 参考资料

《阿里Java开发手册-泰山版》

## 2 程序系统的结构





## 

## 3 客户端前端设计说明

### 3.1 程序描述

本程序是一款名为“番茄清单”的个人效率管理软件，设计目的是帮助用户提高工作和学习效率，通过结合番茄工作法（Pomodoro Technique）和待办事项管理功能，让用户能够更好地规划和跟踪任务进度。该程序的主要特点如下：

主程序与子程序结构：程序包含主程序和多个子程序，如番茄计时器模块、待办事项管理模块、统计分析模块等，各模块之间通过接口进行通信和数据交换。

顺序与并发处理：程序在处理大部分用户界面交互和数据展示时采用顺序处理方式，而在执行后台任务（如计时、数据同步）时采用并发处理，以确保用户界面的流畅性和任务处理的高效性。

本客户端前端程序用于向用户暴露图形界面接口，提供面向一般用户的操作方式。以下为程序关键特点信息：

|  |  |  |
| --- | --- | --- |
| 特点 | 值 | 说明 |
| 常驻 | 否 | 界面模块在隐藏时应释放所有大型资源，以减少硬件负担。 |
| 子程序 | 是 | 本前端程序隶属于系统子程序模块，但相对独立于后端程序。 |
| 处理方式 | 顺序、并发 | 交互使用顺序处理，后台任务使用并发处理 |

### 3.2 功能

3.2.1用户注册

程序接受用户提交的注册信息。包括用户名、账号，密码。为了确保信息安全性和有效性，某些字段可能会有特定的要求，比如长度限制、字符类型要求等。接收到注册信息之后，程序需要执行一系列步骤来验证其合法性。先对提供的用户名，账号和密码进行检查，判断是否符合预设格式规则以及该用户是否存在。若注册信息合法，则用户注册成功，并输出用户信息，跳转到登录页面。若注册信息不合规范，则会返回失败以及不合规范之处，以便让用户知道哪里出了问题，进而纠正错误重新尝试注册。

3.2.2 用户登录

程序接收用户提交的登录信息，包括账号，密码。接受到信息后程序执行一系列步骤对其进行检测，判断账号密码是否符合。若登录信息合法则用户登录成功，跳转到主页面。若登录信息不合法则返回失败。

3.2.3 待办创建与管理

该功能由“最近待办”与“待办详情”两个页面实现。

1）“最近待办”页面

（1）待办列表显示

程序将显示一个列表展示待办事项。若用户此前已创建过待办事项，则列表 将显示用户已创建的所有未删除的待办事项；若用户为首次登录，则列表显 示为空。

（2）待办事项创建

用户可根据输入框中的提示信息创建一个新的待办事项。用户输入待办事项 标题到输入框并按回车。若标题为空则该待办事项不会创建；若不为空则事 项列表会添加新创建的待办事项。

（3）标记待办完成

用户可将待办事项标记为已完成。通过点击待办事项项头的方框，被点击的 待办事项将被标记为“已完成”，并增加“划横线”效果。

（4）删除完成待办

当待办事项已被勾选，即处于“已完成”状态，点击某一待办事项项尾的减 号按钮，即可从待办事项列表中删除该事项。

（5）查看待办详情

用户可查看待办事项的详细信息。通过点击某一待办事项项尾的详情按钮， 程序将跳转至该待办事项的详情页，加载并显示该事项的详细信息。

2）“待办详情”页面

（1）修改标题

通过在输入框中输入待办事项的名称对待办事项的名称进行修改。

（2）编辑待办事项描述

在输入框中输入待办事项的相关描述，对事项内容进行补充。

（3）设置待办事件开始时间

点击按钮选择待办事项的开始时间，可以选择“今天”“明天”和在日历中 自定义事项的开始时间的日期。

（4）设置待办事项结束时间

点击按钮选择待办事项的结束时间，可以选择“今天”“明天”或在日历中 自定义事项的结束时间的日期。

（5）选择待办事项任务的重要程度

点击下拉列表，选择事项的重要程度，分为“不重要”“较不重要”“比较重 要”“很重要”四个等级。

（6）选择待办事项任务的紧急程度

点击下拉列表，选择事项的紧急程度，分为“不紧急”“较不紧急”“比较紧 急”“很紧急”。

3.2.4 番茄钟创建与管理

（1）番茄钟创建：

番茄专注默认时间25分钟，默认休息时间5分钟。用户可自行设定专注与 休息时间，其中专注时间限制为0到60分钟，休息时间限制为0到20分钟。

开始/结束专注命令：用户点击“开始专注”按钮启动倒计时，点击“放弃专 注”按钮结束当前专注周期。

查看记录请求：用户选择查看过去的专注记录。

（2）番茄钟处理：

启动倒计时：根据用户设定的时间开始倒计时。当倒计时结束时，切换到休 息状态并等待用户确认继续下一个专注周期。

暂停与恢复专注：用户在专注期间可以暂时停止倒计时，并在之后恢复倒计 时。

修改专注时间：允许用户更改默认的专注时间长度。修改后保存新的专注时 间设置。

记录与展示专注数据：记录每次专注开始和结束时间、专注时长等信息。根 据用户选择展示专注记录。

（3）番茄钟显示：

显示倒计时：实时显示剩余专注时间。

提示信息：专注开始时提供声音或震动提示。专注结束时通知用户进入休息 阶段。提示用户完成休息并准备进行下一轮专注。

专注历史记录展示：显示选定日期内所有已完成的专注时段列表。展示每个 专注时段的具体细节，例如专注目标、起止时间及专注时长等。

界面反馈：界面元素状态更新，如按钮禁用/启用、对话框弹出与关闭等。

3.2.5用户注销

在当前设备上退出账号，在退出时向云端服务器发送同步请求，将本地数据 同步，确保多端数据统一。

3.2.6云端同步

将用户的信息发送给云端服务器进行存储，实现云端保存和数据同步功能， 保障数据安全。用户可在登录新设备时从云端拉取数据，实现多端设备数据 集同步功能。

### 3.3 性能

1）高精度：计时器模块应具有高精度，误差不超过1秒。

2）灵活性：用户应能够自由添加、修改和删除待办事项，以及自定义番茄工作法的计时参数（如专注时间和休息时间）。

3）即时性：程序响应用户操作的速度应快，界面切换和数据加载的时间不超过1秒。同时，待办事项的搜索和排序功能应能在大规模数据集上实现快速查询。

**3.4 输入项**

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 内容 | 类型 | 格式 |
| 登录注册 | 用户名 | 字符串 | 可见的Unicode字符 |
| 账号 | 字符串 | 可见的Unicode字符 |
| 密码 | 字符串 | 可见的ASCII字符，并且长度在8以上，同时必须包含大小写字母数字和特殊符号 |
| 待办事项 | 待办事项标题 | 字符串 | 可见的Unicode字符 |
| 待办事项副标题 | 字符串 | 可见的Unicode字符 |
| 待办事项描述 | 字符串 | 可见的Unicode字符 |
| 待办事项创建时间 | 数字 | 系统自动确定 |
| 待办事项开始时间 | 数字 | 图形化输入 |
| 待办事项结束时间 | 数字 | 图形化输入 |
| 待办事项重要程度 | 数字 | 图形化输入 |
| 待办事项紧急程度 | 数字 | 图形化输入 |
| 番茄钟 | 专注时间 | 数字 | 图形化输入 |
| 休息时间 | 数字 | 图形化输入 |

### 3.5 输出项

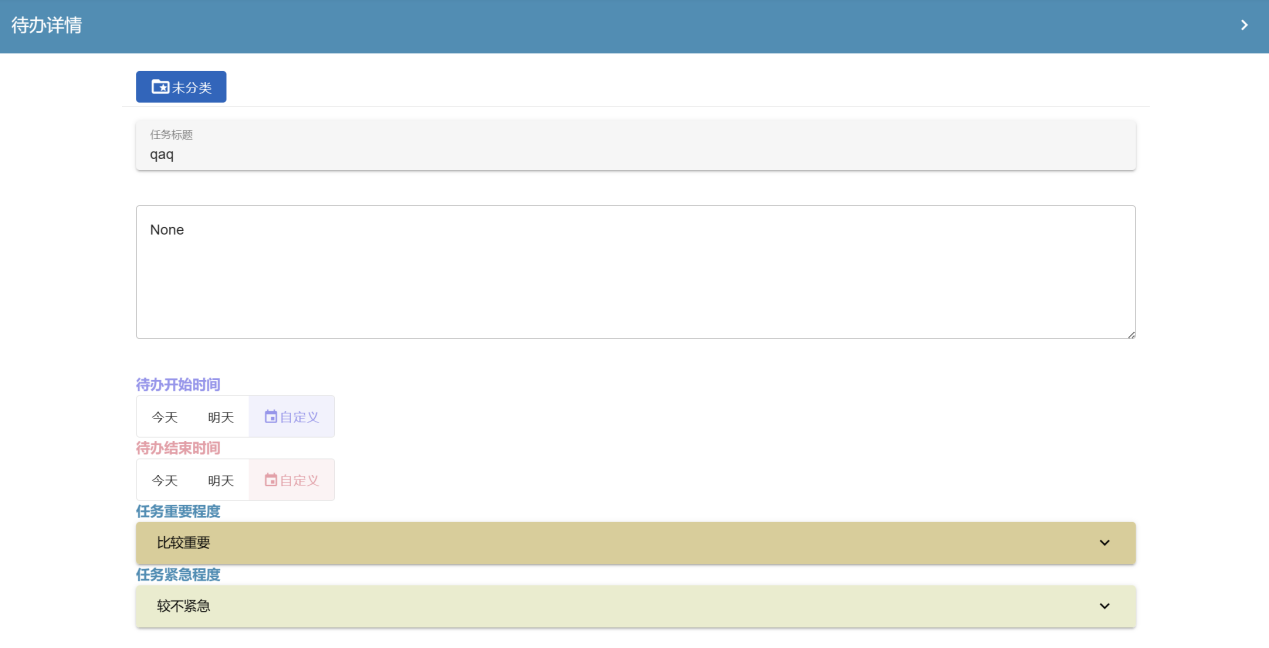
1. 待办事项列表

在“最近待办”页面呈现，一行呈现一件待办事项，显示待办事项的标题和备注。右侧包含三个按钮，功能分别为开始番茄钟、显示待办详情和删除。



1. 待办详情

在待办事件列表，点击某事件的显示待办详情按钮后呈现。任务标题使用文字框显示，任务备注使用文字域显示，待办开始时间、待办结束时间通过按钮组显示，任务重要程度、任务紧急程度通过下拉栏显示。



1. 番茄时钟

在待办事件列表，点击某事件的开始番茄钟按钮后呈现。输入时间后，会实时显示剩余时间，完成后会弹窗通知。

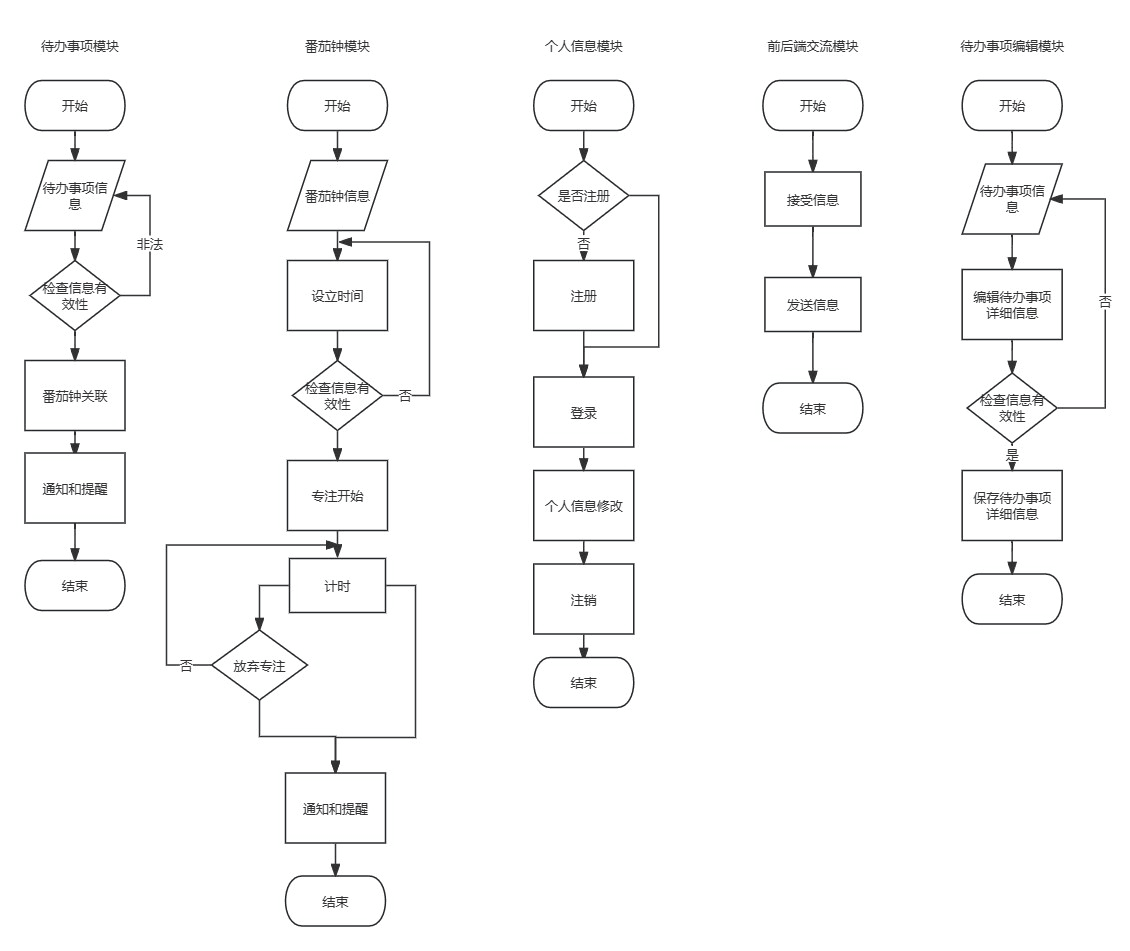


### 3.6 算法

本程序未使用较复杂的、外部引入的算法。

**3.7 流程逻辑**

该应用的主要功能包括如图所示几个部分。



3.7.1个人信息模块

（1）用户启动应用程序。

（2）如果用户未登录，显示登录/注册界面。用户可以选择注册或登录。

（3）如果选择注册：

1. 用户输入注册信息（账号、用户名、密码）。
2. 系统验证输入信息的有效性。
3. 如果信息有效，系统创建新用户账户并提示用户注册成功。
4. 跳转回登录界面。

如果选择登录：

1. 用户输入用户名和密码。
2. 系统验证用户名和密码。
3. 如果验证成功，用户登录并进入主界面。
4. 如果验证失败，显示错误消息并允许用户重新输入。

3.7.1待办事项模块：

（1）用户进入“最近待办”页面，程序将显示一个列表和一个输入框，列表用于展示待办事项，输入框用于创建新的待办。

（2）如果用户此前已创建过待办事项，则程序会根据用户名从后端拉取其待办事项，列表将显示该账号中已创建的所有未删除的待办事项；如果用户为首次登录，则列表显示为空。

（3）当用户将待办事项的标题输入到输入框中并按下回车键时，程序会检测待办事项的标题是否为空。若标题为空则该待办事项不会创建；若不为空则程序会向后端发送创建该待办事项的请求，同时本地网页的待办事项列表中会添加新创建的待办事项。

（4）通过点击待办事项项头的方框，程序会将被点击的待办事项标记为“已完成”，并增加“划横线”效果。

（5）当待办事项已被勾选，即处于“已完成”状态，点击某一待办事项项尾的减号按钮，程序就会向后端发送删除该待办事项的请求，同时本地网页的待办事项列表中会删除该待办事项。

（6）通过点击某一待办事项项尾的详情按钮，程序会显示该事项的详细信息。

（7）每个待办事项设有独立的番茄钟，点击某一待办事项项尾的番茄钟按钮，程序将跳转到对应事项的番茄钟页面。

3.7.1待办事项编辑模块：

用户可以在待办事项界面上访问待办详情模块。

用户可执行以下操作：

1）选择待办任务分类：选择当前待办事项所归属的文件夹（文件夹由用户自行设置，便于管理类型相似的某一类待办），也可以不选择分类的文件夹。

2）编辑待办任务标题：在输入框中输入待办事项的任务名，即可为当前待办设置名称。

3）编辑待办任务备注：在输入框中输入对待办任务的描述以及所要注意的细节。

4）选择待办事项的开始时间：可以点击按钮选择待办任务的开始时间。

5）选择待办事项的完成时间：可以点击按钮选择待办任务的截止时间。

6）选择待办事项的紧急程度：待办事项的紧急程度分为四种：不紧急、较不紧急、较紧急、紧急。

7）选择待办事项的重要程度：待办事项的重要程度分为四种：不重要、较不重要、较重要、重要。

8）删除待办事项：选择要删除的待办事项并确认删除操作。

3.7.1番茄钟模块：

用户在主界面上访问番茄钟模块。

用户可执行以下操作：

1）开始新的番茄钟：设置专注时间和休息时间，点击“开始”按钮启动番茄钟计时。

2）暂停当前番茄钟：在计时期间，用户可以暂停番茄钟。

3）继续当前番茄钟：在暂停状态下，用户可以继续进行剩余的计时。

4）结束当前番茄钟：在专注期或休息期结束后，用户可以选择结束当前番茄钟。

3.7.2数据统计模块：

用户在主界面上访问数据统计模块。

系统根据用户的活动记录自动分析以下数据：

番茄钟数据：总番茄钟数等。

待办事项与完成数据：总待办事项数、已完成事项数、完成率等。

系统将分析结果以列表的形式展示给用户，帮助用户了解自己的工作效率和进步情况。

3.7.3前后端交互模块：

用户通过各种界面元素（如按钮、输入框、滑块等）与应用程序进行交互。

前端监听用户的操作事件，根据事件类型调用封装的函数或执行本地逻辑。

后端处理请求后返回数据或状态更新给前端；前端接收到响应后更新界面内容，向用户反馈操作结果。

### 3.8 接口

前端需要与后端进行通信，为方便调用，对接口进行了封装。封装函数分为待办事项、番茄钟、用户三类。

1. 待办事项方法

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 方法名 | 参数 | 返回值 |
| ToDoWorkRequest | create | 待办事项（对象） | 属性content值为新创建的待办事项innerId |
| query | 待查询待办事项innerId（数字） | 待办事项（字符串） |
| delete | 待删除待办事项innerId（数字） | 空 |
| edit | 待办事项（对象） | 空 |
| enumerate | 空 | 所有待办事项（嵌套字符串） |
| PomodoroRequest | start | 待绑定待办事项innerId（数字） | 空 |
| end | 空 | 空 |
| edit | PomodoroInfo（对象） | 空 |
| AccountRequest | useLogin | 账号，密码 | 空 |
| useRegister | 用户名，账号，密码 | 空 |

1. 番茄钟方法
2. 用户方法

### 3.9 存储分配

前端初始化时不存储任何数据，通过前后端通讯获取数据。

待办事项列表在本地存储，位置为D盘/test/list.txt。

待办事项详情、番茄钟、用户数据等其他数据均不进行本地存储，按需向后端请求。

### 3.10 注释设计

为确保代码可读性，本项目的注释设计方案依照阿里巴巴前端开发注释规范设计。

### 3.11 限制条件

**网络连接：**

用户在使用用户账号登录、注册功能、数据同步功能时，必须有可靠的网络连接支持才能连接。同时，服务器端应保持良好的网络环境，保障用户的使用体验。

### 3.12 测试计划

#### 3.12.1. 测试技术要求

单元测试：对每个功能模块（如用户注册、登录、待办事项管理、番茄钟等）进行独立的单元测试，确保各模块内部逻辑正确无误。

集成测试：在单元测试通过后，将各个模块集成起来进行测试，以检测不同模块间的接口调用和数据交互是否正常。

UI测试：针对前端界面进行操作验证，确保按钮点击、页面跳转、表单提交等功能表现符合设计需求。

边界值分析：对输入项的数据范围进行边界值测试，包括最小值、最大值以及接近边界值的情况。

异常测试：模拟非法或不符合规范的输入，检查程序能否给出正确的错误提示及处理机制。

3.12.2 输入数据与预期结果

用户注册：

输入有效数据，预期结果是成功创建用户并跳转至登录页面。

输入无效数据（如密码格式错误、用户名已存在），预期结果是返回相应的错误提示信息。

用户登录：

输入有效账号密码，预期结果为登录成功并跳转到主页面。

输入无效账号密码，预期结果是登录失败并显示错误信息。

待办创建与管理：

正确创建待办事项，预期结果是在列表中添加新事项。

标记完成、删除、查看详情等操作，预期结果分别是事项状态更新、从列表移除及跳转至详情页面并展示详细信息。

番茄钟创建与管理：

创建和修改专注时间，预期结果是倒计时开始/暂停/恢复功能正常，并能记录专注数据。

查看专注历史记录，预期结果是正确显示历史专注时段。

用户注销与云端同步：

注销时，预期结果是本地数据同步至云端且用户退出登录状态。

登录新设备时，预期结果是从云端拉取数据，实现多端数据同步。

3.12.3 进度安排

第一阶段（1周）：编写和执行单元测试脚本，每位成员负责各自模块；

第二阶段（1周）：集成测试，由前后端交叉协作完成；

第三阶段（1周）：UI和兼容性测试，确保不同浏览器和设备上的表现一致；

第四阶段（1周）：压力测试和性能测试，检验系统在大规模数据集下的响应速度及稳定性；

第五阶段（持续）：回归测试，在每次代码更改后重新运行测试套件，保证原有功能不受影响。

3.12.4 人员职责

前端组：负责前端组件的单元测试、UI测试及跨平台兼容性测试；

后端组：负责服务器接口、业务逻辑和数据库交互的单元测试及集成测试；

服务器负责人：确保服务器环境稳定，测试数据同步功能及相关性能指标。

3.12.5 设备条件与驱动程序

搭建测试环境，准备多种设备（PC、移动设备）和浏览器版本；

对于API接口测试。

3.12.6 桩模块

在部分依赖未完成或无法直接测试的情况下，编写桩模块模拟依赖行为，确保被测试模块可以独立验证。

3.12.7 安全保密条件

所有涉及用户敏感信息的测试应遵循严格的隐私政策，采用匿名或伪造数据；

测试过程中的数据交换需加密传输，确保信息安全；

测试完成后清理测试数据，避免信息泄露。

### 3.13 尚未解决的问题

在当前阶段，经过团队成员的共同努力和细致开发，我们已成功实现了用户注册与登录、待办事项创建与管理、番茄钟功能以及云端同步等核心功能。所有模块均按照预定设计规格进行开发，并通过了详尽的单体测试和集成测试。在此过程中，针对各项功能的输入验证、逻辑处理、数据持久化存储、界面展示等方面进行了严格的质量把控。

目前，本系统已能满足用户在多设备间进行数据同步的需求，同时提供了灵活的待办事项管理和自定义番茄工作法计时参数等功能，确保了高精度的时间计算和即时响应速度。此外，在安全保密方面，我们也采取了相应的措施，确保用户敏感信息的安全存储与传输。

基于上述情况，现阶段本程序暂未发现影响功能实现或用户体验的重大问题。然而，作为一款持续迭代的产品，我们仍然关注于性能优化、用户体验提升、新功能需求收集与实施等问题，以便在未来版本中进一步完善和增强系统的整体表现和功能丰富度。同时，我们会密切关注上线后用户反馈，及时修复可能出现的任何未知问题，以提供更稳定、高效的服务体验。

## 4 客户端后端设计说明

### 4.1 程序描述

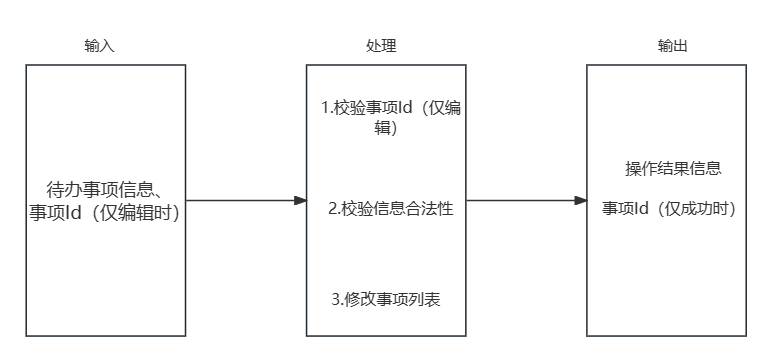
本客户端后端程序为系统主要功能实现主体，为系统提供各主要模块的实际处理过程，并暴露相关访问接口给本地客户端前端程序。以下为本子程序关键信息：

|  |  |  |
| --- | --- | --- |
| 特点 | 值 | 说明 |
| 常驻 | 是 | 核心程序基于控制台，占用资源较少，因此常驻以提供更连续的体验。 |
| 子程序 | 是 | 本后端程序为系统的主体功能实现的子程序。 |
| 可重入 | 否 | 后端程序只允许存在一个。 |
| 处理方式 | 顺序+并发 | 内部处理通常使用顺序处理，但考虑到可能有多个前端程序存在，需要处理多线程连接的并发处理情况。 |

### 4.2 功能

4.2.1 待办事项的创建和编辑

文字描述与本系统需求说明书中对应文档相同，以下为IPO图



4.2.2 待办事项的通知与提醒

本模块主要用于通知用户待办事项的 开始、超时与可能的遗忘。

通知的逻辑如下：

由于待办事项有时间顺序，在单个时间点内只需要对最近的事项进行检测即可。当该事件完成后，关注的事件按时间排序向后推移，检测下一个事件的通知事件。

进行通知的类型有以下几个：

* 待办事项开始通知
* 待办事项防遗忘通知
  + 用户设定的事项预期完成时间前24h时
* 待办事项超时通知
  + 当前时间点超过设定的完成时间而事件没有被标记完成时

具体来说，进入开始状态的契机如下：

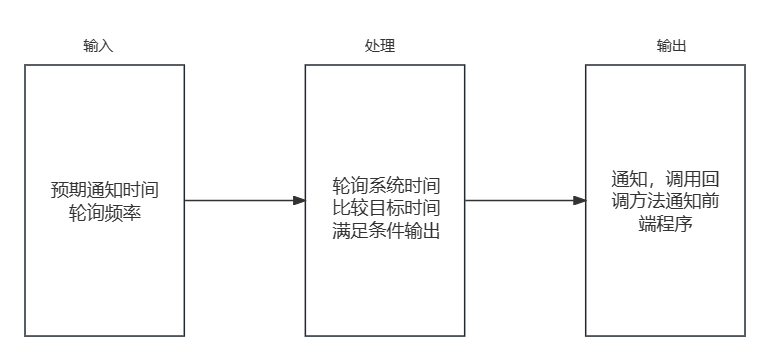
a.根据用户设置，到达了事项开始的时间

进入完成状态的契机如下：

a.用户将该事项标记为完成状态

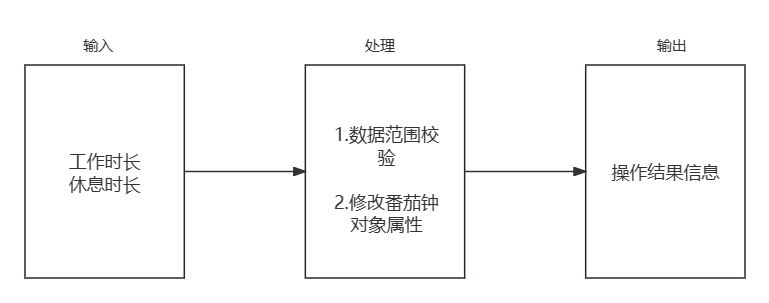
进入超时状态的契机如下：

a.待办事项到达预期的完成时间，但是用户没有把事项标记为完成状态



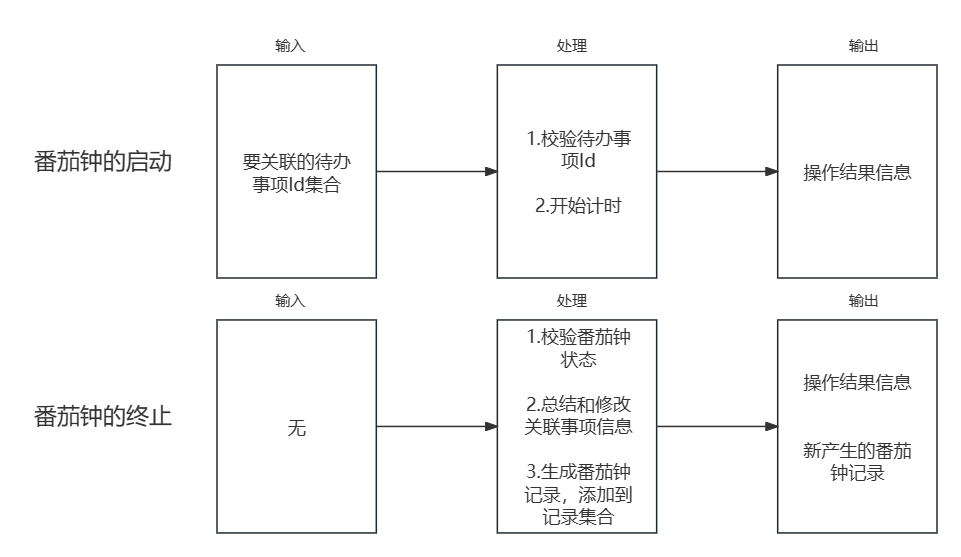
4.2.3 番茄钟编辑

本模块用于进行编辑番茄钟的信息，逻辑较为简单，如下IPO图。



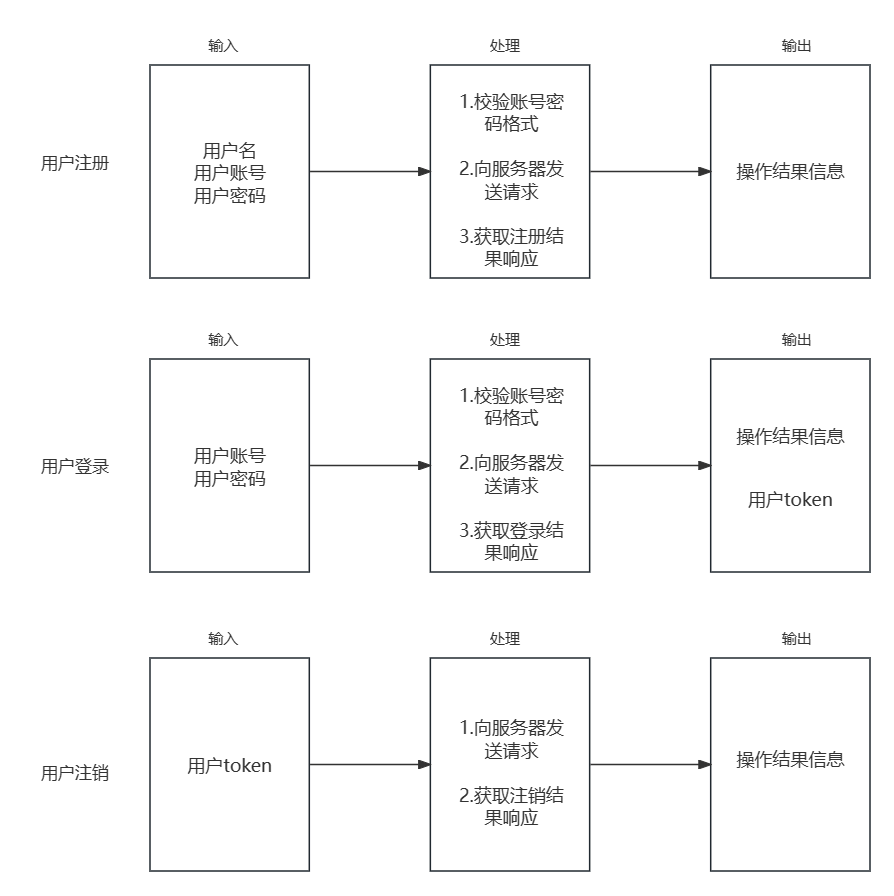
4.2.4 番茄钟使用

本模块用来启动番茄钟计时。

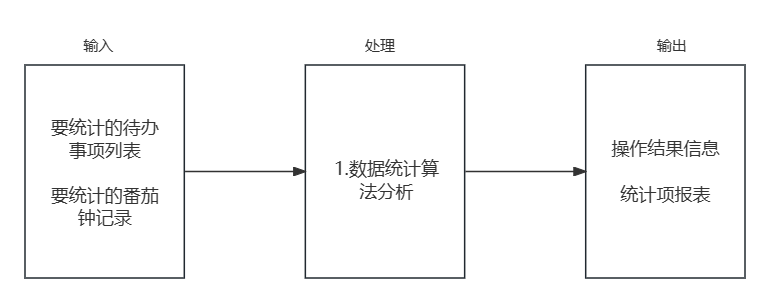


4.2.5 用户的注册、登录与注销

本模块为同步模块提供用户验证功能。



4.2.6 数据统计模块



### 4.3 性能

时间特性：用户打开本地后端程序的时间应小于等于2秒。如果可能，应通过包括数据缓存、后台程序的方式尽量减少页面显示时间。实现时，应在资源占用和启动时间之间寻得平衡。

### 4.4 输入项

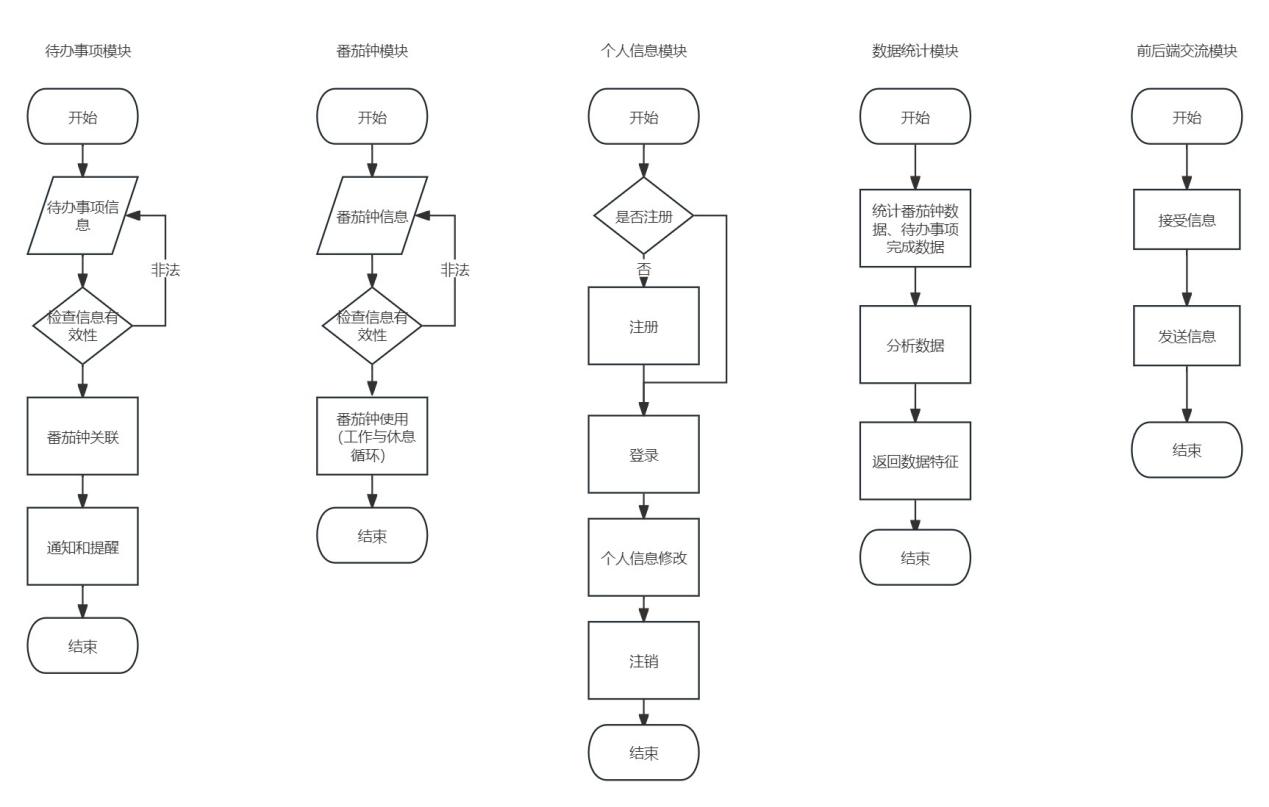
|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 内容 | 类型 | 数量 |
| 待办事项修改请求 | 操作类型 | 枚举 | 1 |
| 待办事项信息 | 待办事项 | 1 |
| 待办事项ID | int | 1 |
| 待办事项查询请求 | 操作类型 | 枚举 | 1 |
| 待办事项ID | int | 1 |
| 待办事项创建请求 | 操作类型 | 枚举 | 1 |
| 待办事项信息 | 待办事项 | 1 |
| 枚举所有待办事项请求 | 操作类型 | 枚举 | 1 |
| 删除待办事项请求 | 操作类型 | 枚举 | 1 |
|  | 待办事项ID | int | 1 |
| 开始番茄钟计时请求 | 操作类型 | 枚举 | 1 |
| 绑定待办事项ID | int | 1 |
| 结束番茄钟计时请求 | 操作类型 | 枚举 | 1 |
| 用户登录请求 | 操作类型 | 枚举 | 1 |
| 用户登录信息 | String | 1 |
| 用户登出请求 | 操作类型 | 枚举 | 1 |
| 用户身份凭证 | String | 1 |
| 用户注册请求 | 操作类型 | 枚举 | 1 |
| 注册信息 | String | 1 |
| 同步请求 | 操作类型 | 枚举 | 1 |
| 用户数据 | String | 1 |

### 4.5 输出项

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 内容 | 类型 | 数量 |
| 操作结果信息 | 事项ID（仅成功时） | int | 1 |
| 通知 | 调用回调方法 |  | 1 |
| 登录结果 | 登录结果信息 | String | 1 |
| 注册结果 | 注册结果信息 | String | 1 |
| 数据展示 | 完成数据信息 | String | 1 |

### 4.6 算法

### 4.7 流程逻辑



### 4.8 接口

### 1）接口1

接口描述：后端与前端数据传输

业务目的：本系统采用前后端程序进程隔离的框架，该接口使后端程序和前端程序进行联系，将后端内容在前端中显示。

使用时机：启动程序后，用户在前端进行各种操作时，前端将数据传输至后端，后端将数据传输至前端进行显示。

使用频率：取决于用户操作频率。峰值时应能满足在100ms以下完成100KB以下的数据传输。

接口特点：因为本程序采用前后端分离，故该接口为前后端信息交流通道。需要实现双向套接字通讯。

内容与格式：主要使用基于Json的序列化和反序列化进行数据传输。

交换过程：每当用户进行操作时，相关请求由前端程序发送给后端程序，后端程序返回操作结果信息。 当特定事件发生时，后端程序将通知内容发送给前端程序。

数据包说明：传输的数据包具有 数据长度、数据类型、数据内容和扩展信息。

协议要求：基于流和TCP/IP的Socket通讯协议。

性能要求：本地通讯的数据传输接口的响应时间通常在 20ms以内。

环境限制：操作系统应当允许通过特定端口的套接字通讯。

1. **接口2**

接口描述：后端与服务器数据传输

业务目的：将后端数据上传至服务器、从服务器获取数据，完成用户管理、数据同步功能

使用时机：用户注册、登录、个人信息编辑；数据同步时

使用频率：取决于用户操作频率。在网络环境正常时，应能满足在1s以下完成100KB以下的数据传输。

接口特点：完成后端与服务器数据传输，需要实现单向套接字通讯。

内容与格式：主要使用基于Json的序列化和反序列化进行数据传输。 交换过程：当用户进行用户管理操作和数据同步操作时，相关请求由后端程序发送给服务器，服务器返回操作结果信息。

数据包说明：传输的数据包具有 数据长度、数据类型、数据内容和扩展信息。

协议要求：HTTP通讯协议。

性能要求：公网通讯的数据传输接口的响应时间通常在200ms内。

环境限制：由于应用涉及在线同步和数据存储，需要确保用户能够稳定连接到互联网。变化的互联网连接状态可能影响数据同步和实时性。

### 4.9 存储分配

翱翔清单 App 的后端存储分配旨在支持用户数据的安全存储以及系统性能的优化。以下是主要的后端存储分配策略：

文件存储：

对于用户上传的文件、图片等附件，后端系统采用文件存储系统进行存储。

缓存机制：

为提高系统性能，后端引入缓存机制，通常使用缓存数据库（如Redis）。

安全性和备份：

数据的安全性是关键，因此采用数据加密，对于敏感数据采用加密算法，保 障数据在传输和存储时的安全性。

### 4.10 注释设计

为确保代码可读性，本项目的注释设计方案依照《阿里Java开发手册-泰山版》中的注释规范设计。

### 4.11 限制条件

**操作系统限制：**

用户在使用时，必须根据所使用的系统选择合适的版本。

### 4.12 测试计划

自动化测试工具： 使用适当的自动化测试工具（如Appium、XCUITest、Espresso等）进行功能测试。

各种任务类型： 包括普通任务、紧急任务、重要任务等。

不同日期和时间： 用于测试时间轴视图和日历视图下的任务显示和排序。

不同的番茄钟数据：用于测试番茄钟时间记录和任务关联。

### 4.13 尚未解决的问题

本项目的需求目前的技术均可实现，无尚未完成的问题。

## 5 服务端设计说明

### 5.1 程序描述

本服务端运行于服务器上，用于读写存储用户账号信息及待办事项信息的数据库，并通过HTTP协议与客户端后端进行通讯，接收、处理与响应客户端的注册、登录、注销、修改、拉取、覆写HTTP请求，从而完成数据的云端同步以及与云端同步相关的用户注册、登录、注销、信息修改等功能。

### 5.2 功能

1. 接收、处理、响应注册请求
2. 接收、处理、响应登录请求
3. 接收、处理、响应注销请求
4. 接收、处理、响应修改请求
5. 接收、处理、响应拉取请求
6. 接收、处理、响应覆写请求

### 5.3 性能

前端与后端程序之间的通讯主要采用本地 请求——响应 的模式，效率取决于内存IO速度。只需要考虑通讯数据量大小和响应时间的因素。根据初步估计，单次通讯量通常小于1MB，响应时间应接近即时（任何本地请求应在50ms内完成）。

后端程序在与云服务器进行同步时，采用异步HTTP请求方式，效率取决于网络传输速度。单次通讯量通常小于1MB，预期响应时间不超过若干秒，超过30s时，视为网络传输异常。

### 5.4 输入项

来自客户端后端的HTTP请求：保持监听，收到请求时即为输入的时机。其中密码（包括“密码”“旧密码”“新密码”）均为经过md5映射后的内容。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 来源 | 名称 | 协议 | 内容 | 类型 | 数量 |
| 客户端后端 | 注册请求 | HTTP | 用户名 | 字符串 | 1 |
| 账号 | 字符串 | 1 |
| 密码 | 字符串 | 1 |
| 登录请求 | HTTP | 账号 | 字符串 | 1 |
| 密码 | 字符串 | 1 |
| 注销请求 | HTTP | token | 字符串 | 1 |
| 修改请求 | HTTP | token | 字符串 | 1 |
| 新用户名 | 字符串 | 0-1 |
| 旧密码 | 字符串 | 0-1 |
| 新密码 | 字符串 | 同上 |
| 拉取请求 | HTTP | token | 字符串 | 1 |
| 覆写请求 | HTTP | token | 字符串 | 1 |
| 同步事项列表 | JSON格式字符串 | 1 |

来自数据库的内容：处理需要访问数据库的请求时，从数据库获取数据。

其中“时间”是UNIX纪元时间戳；数量"n"表示在数据库中搜索。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 来源 | 名称 | 内容 | 类型（格式） | 数量 |
| SQLite数据库 | 注册验证 | 账号 | TEXT | n |
| 登录验证 | 账号 | TEXT | n |
| token | TEXT | 1 |
| token过期时间 | TEXT | 1 |
| 密码 | TEXT | 1 |
| 注销验证 | token | TEXT | n |
| token过期时间 | BIGINT | 1 |
| 修改验证 | token | TEXT | n |
| token过期时间 | BIGINT | 1 |
| 旧密码 | TEXT | 1 |
| 拉取验证 | token | TEXT | n |
| token过期时间 | BIGINT | 1 |
| 同步事项列表 | TEXT（JSON） | 1 |
| 覆写验证 | token | TEXT | n |
| token过期时间 | BIGINT | 1 |

### 5.5 输出项

向客户端后端回复HTTP响应：处理完成后响应客户端的HTTP请求。

成功："Success"

失败："Failure"

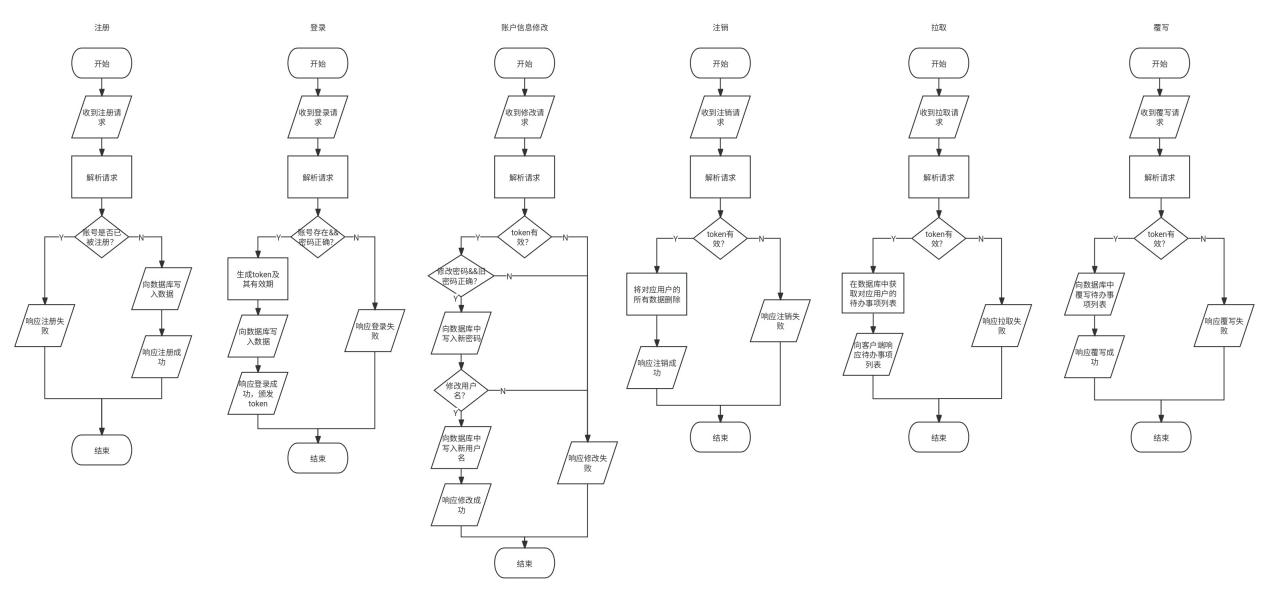
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 目的 | 名称 | 协议 | 内容 | 类型 | 数量 |
| 客户端后端 | 注册响应 | HTTP | 成功/失败状态 | 字符串 | 1 |
| 登录响应 | HTTP | token | 字符串 | 1 |
| 成功/失败状态（成功返回token，失败返回空） | 字符串 | 1 |
| 注销响应 | HTTP | 成功/失败状态 | 字符串 | 1 |
| 修改响应 | HTTP | 成功/失败状态 | 字符串 | 1 |
| 拉取响应 | HTTP | 同步事项列表（成功）/失败 | JSON字符串/字符串 | 1 |
| 覆写响应 | HTTP | 成功/失败状态 | 字符串 | 1 |

写入数据库：处理需要向数据库中写入数据的请求时，将数据写入数据库。

其中“时间”是UNIX纪元时间戳。-1代表在数据库中将该项数据标记为空。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 目的 | 名称 | 内容 | 类型（格式） | 数量 |
| SQLite数据库 | 注册登记 | 账号 | TEXT | 1 |
| 密码 | TEXT | 1 |
| 用户名 | TEXT | 1 |
| 登录登记 | token | TEXT | 1 |
| token过期时间 | TEXT | 1 |
| 注销登记 | token | TEXT | -1 |
| token过期时间 | BIGINT | -1 |
| 修改登记 | 用户名 | TEXT | 1 |
| 新密码 | TEXT | 1 |
| 拉取待办 | / | / | / |
| 覆写待办 | 同步事项列表 | TEXT（JSON） | 1 |

### 5.6 流程逻辑



### 5.7 接口

1. 与客户端的通讯采用HTTP协议，接收、处理、响应来自客户端的HTTP请求。
2. 处理数据的Servlet由Tomcat服务器调用。Tomcat在接收到请求后，会根据请求的URL调用相应的Servlet，从而处理不同的请求。请求处理完成后的响应也由Servlet通过Tomcat服务器发送至客户端。
3. 数据库引擎为SQLite，Servlet通过SQLite JDBC访问数据库。

### 1）接口1

接口描述：后端与服务器数据传输

业务目的：将后端数据上传至服务器、从服务器获取数据，完成用户管理、数据同步功能

使用时机：用户注册、登录、注销、个人信息编辑；数据同步时

使用频率：取决于用户操作频率。在网络环境正常时，应能满足在1s以下完成100KB以下的数据传输。

接口特点：完成后端与服务器数据传输，需要实现双向HTTP通讯。

内容与格式：主要使用基于Json的序列化和反序列化进行数据传输。

交换过程：当用户进行用户管理操作和数据同步操作时，相关请求由后端程序发送给服务器，服务器返回操作结果信息。

数据包说明：传输的数据包具有 数据长度、数据类型、数据内容和扩展信息。

协议要求：HTTP通讯协议。

性能要求：公网通讯的数据传输接口的响应时间通常在200ms内。

环境限制：由于应用涉及在线同步和数据存储，需要确保用户能够稳定连接到互联网。变化的互联网连接状态可能影响数据同步和实时性。

**2）接口2**

接口描述：Tomcat服务器调用servlet

业务目的：接收与响应来自客户端的请求，处理请求对象

使用时机：用户注册、登录、注销、个人信息编辑；数据同步时

使用频率：取决于用户操作频率。

内容与格式：HTTPServletRequest与HTTPServletResponse类Java对象。

交换过程：接收到请求时，Tomcat根据请求URL调用对应的servlet处理请求，而后将处理结果返回给Tomcat。

协议要求：servlet为HTTPServlet的子类。

性能要求：性能由Tomcat的性能以及服务器的性能决定，没有太多调整空间。一般能够满足响应的性能要求。

环境限制：需要在服务器本地具有高于或等于JRE17环境。

**3）接口3**

接口描述：servlet访问数据库

业务目的：将用户数据保存于数据库中，进行增删改查

使用时机：用户注册、登录、注销、个人信息编辑；数据同步时

使用频率：取决于用户操作频率。

内容与格式：SQLite操作。

交换过程：需要访问数据库时，servlet调用JDBC包中的函数进行访问。

性能要求：由SQLite的性能，JDBC库函数的性能与服务器硬件性能决定。一般能够满足读写数据库的性能要求。

环境限制：需要在服务器本地具有高于或等于JRE17环境。

### 5.8 存储分配

翱翔清单 App 的服务端存储分配旨在支持用户数据的安全存储、数据的同步以及系统性能的优化。以下是主要的服务端存储分配策略：

1）数据库存储：

服务端采用关系型数据库（例如 MySQL、PostgreSQL）或 NoSQL 数据库（例如 MongoDB）来存储用户数据。

2）缓存机制：

为提高系统性能和降低数据库访问频率，服务端引入缓存机制，通常使用内存缓存或缓存数据库（如Redis）。

3）安全性和备份：

服务端数据的安全性是首要考虑因素，因此采用以下策略：

- 数据加密：对于敏感数据（如用户密码）采用合适的加密算法存储，不直接存储明文，确保数据在传输和存储时的安全性。

- 定期备份：定期对数据库和文件存储进行备份，以防止数据丢失和提高系统可恢复性。

### 5.9 注释设计

为确保代码可读性，本项目的注释设计方案依照《阿里Java开发手册-泰山版》中的注释规范设计。

### 5.10 限制条件

**网络连接：**

用户在使用用户账号登录、注册功能、数据同步功能时，必须有可靠的网络连接支持才能连接。同时，服务器端应保持良好的网络环境，保障用户的使用体验。

### 5.11 测试计划

**技术要求：**

自动化测试工具：使用适当的自动化测试工具（如Appium、XCUITest、Espresso等）进行功能测试。

性能测试工具：使用性能测试工具（如JMeter、LoadRunner等）进行性能测试，模拟多用户、高负载场景。

**输入数据：**

数据同步测试：用于数据同步功能。

**预期结果：**

所有模块均能正常响应用户操作，无任何严重错误，数据在不同的设备之间可以及时同步。

### 5.12 尚未解决的问题

本项目的需求目前的技术均可实现，无尚未完成的问题。