队伍成员

史天正，余悦，董馨忆

小组成员分工情况

史天正：想法设计，后端(数据库的内容存储读取，主要在utils.cpp,utils.h提供了标准格式的文档)

余悦：统计模块，提醒模块

董馨忆：分组模块、时间轴模块

项目介绍

程序功能介绍

我们的基本功能包括：

* 提醒：当用户使用某个应用分组的时间达到指定的使用时间时，程序会提醒用户自定义内容，以便用户可以更好地控制自己的应用程序使用时间。
  + 增加提醒并设置提醒时间限制
  + 删除提醒
  + 在提醒时间到后显示提醒消息
* 时间轴：时间轴可以展示用户一天之内使用应用的时间轴，以便用户更好地了解自己的应用程序使用情况，每天在哪个时间段更容易使用哪种应用。
* 统计：用户可以选择指定的开始日期和结束日期，以显示在该时间段内使用软件的时长统计图，以便用户更好地了解自己的应用程序使用情况。
* 分组：用户可以将使用的软件进行分组，以便更好地管理和控制自己的应用程序。
  + 增加分组
  + 双击分组名称重命名
  + 删除分组
  + 双击应用名更改其分组

项目各模块与类设计细节

我们采用前端 + 后端式架构。整个系统分为两大部分：前端与后端。后端负责维护、保存、处理数据，前端则负责提供图形界面，与用户交互。

前端设计

基本窗口

使用一个Tab，可以切换展示时间轴、统计、分组操作、提醒操作四个内容

分组

* InitTreeWidget：读取数据库已有分类并显示类别下的应用名，用treewidget实现
* 删除类别(on\_pushButton\_2\_clicked()）：点击类别名后删除类别，会弹出确认窗口，如确认则类别删除，其下应用会移动到第一个类别下
* 添加类别(on\_pushButton\_clicked())：点击添加类别并输入类别名可添加类别，若类别已存在会显示错误
* 重命名类别(on\_CategoryList\_doubleClicked)：双击类别名字可以重命名类别，若改名字类别已存在或名字与原名相同显示错误
* 更改分组(onAppItemDoubleClicked)：双击应用名称并输入新的类别名可以更改分组，如果类别不存在则提示错误并要求重新输入

时间轴

* CreateTimeline：读取时间轴信息并建立时间轴，用方块表示每个小时使用应用的情况，高度与使用时间成正比，并在下方建立图例，以区别不同应用
* Generatecolor：根据应用名称生成各不相同的颜色以制作图例，并调整了rgb的偏移量使得颜色相差尽量大，容易区分

提醒

* 展示提醒（displayReminders）: 从数据库读取所有的reminder并将其显示在界面中，从左到右为四列，分别展示reminder的name，BindCategory，TimeLimit和Message
* 新增提醒（myaddreminder）：点击add reminder按键后会弹出myaddreminder窗口。在其中输入reminder的name，BindCategory，TimeLimit和Message后点击添加按钮即可弹出Reminder added successfully信息提示窗口；点击return按钮则会返回原界面。
* 删除提醒（mydeletereminder）：点击提醒框中某一具体提醒后，即会出现delete reminder按钮（在未选中时或刷新后不显示），此时点击delete reminder按钮，即会弹出确认窗口，确认后可成功删除。
* 刷新提醒：在进行删除或添加操作后，可进行刷新查看当前所有reminder，且delete reminder键会消失。展示提醒（displayReminders）: 从数据库读取所有的reminder并将其显示在界面中，从左到右为四列，分别展示reminder的name，BindCategory，TimeLimit和Message

统计

* 展示时间统计（createtimesum()）：在主窗口初始化时进行展示，默认设定起止时间均为当天，读取起止时间内各app的名称和使用时长并显示在界面中。
* 更改起止时间：使用dateEdit进行展示和更改查询的起止时间（初始化均为当天），每次更改后重新展示统计信息。

后端设计

- 数据库使用sqlite3文件，包含表：

- HourLog：记录每天的日期、时间（以分钟为单位）和应用程序。

- date INTEGER(YYYYMMDDHH)

- time INTERER (in seconds)

- \*\*app INTERER\*\*

- AppModels：记录应用程序的名称、文件和类别ID。

- \*\*app INTERER PRIMARY KEY\*\*

- file NVARCHAR

- name NVARCHAR \*默认从file中读入去除拓展名的文件名,

- CategoryID int NULL DEFAULT 0

- CategoryModels: 分类ID-名称对照

- \*\*CategoryID INTEGER PRIMARY KEY\*\*

- name NVARCHAR

- ReminderModels

- ReminderID INTEGER PRIMARY KEY

- ReminderName NVARCHAR

- ReminderBindCategory int

- ReminderTimeLimit int //time limit as in minutes within a day

- ReminderMessage NVARCHAR

- 界面部分：【完整的类型和说明可见utils.h】

- 分组：用户可以将使用的软件进行分组，以便更好地管理和控制自己的应用程序。

- 分组

- get\_all\_category

- rename\_category

- add\_category

- delete\_category

- 应用

- get\_apps\_under\_category

- move\_app\_to\_category

- 提醒：当用户使用某个应用分组的时间达到指定的使用时间时，程序会提醒用户自定义内容，以便用户可以更好地控制自己的应用程序使用时间。

- struct Reminder {

int reminderID;

std::string reminderName;

int reminderBindCategory;

int reminderTimeLimit;

std::string reminderMessage;

sqlite3 \*db;

bool reached();

};

- get\_all\_reminder

- add\_reminder

- remove\_reminder

- modify\_reminder

- 统计：用户可以选择指定的开始日期和结束日期，以显示在该时间段内使用软件的时长统计图，以便用户更好地了解自己的应用程序使用情况。

- get\_usage\_app\_timespan

- 时间轴：时间轴可以展示用户在指定日期内使用进程的时间轴，以便用户更好地了解自己的应用程序使用情况。

- get\_app\_usage\_timeline\_day

- 后台进程部分：

- 记录进程：每10秒获取Windows当前focus的窗口进程名，并将其记录到数据库中，以便用户可以更好地了解自己的应用程序使用情况。 数据库：

项目总结与反思

由于时间问题，还有一些设想的功能未实现，比如在关闭主窗口后缩小到托盘；本想用拖拽的方式实现应用更换分组，更加方便快捷；界面较为简单，还未能完成美化；时间轴查看较为困难，希望能找到更直观的方式展示一天之中使用应用的频率；希望能实现常见应用和分类的预置数据库；希望能实现浏览器具体网页的追踪监控；希望能使用特定账号存储数据。