

# 软件工程实验报告

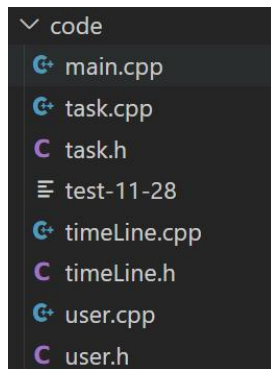
## ——实验三：软件实现与构建

221220130 张浩

此次实验主要从实验二的 UML 图出发,借助 ChatGPT 分析 UML 图中各模块之间的联系并生成代码框架,在此基础上根据软件实际需求对代码框架进行修改和补充,最终完成了一个代码规模为 655 行的任务单 APP 框架,可以满足人们对日常任务计划的基本需求,例如新建任务和编辑已有任务、在自定义的任务提醒时间前向用户发出通知、对任务进行分类并支持为各种类添加标签、任务时间线视图等,同时也支持任务优先级设置和任务进度追踪等功能。UI 界面及目前代码框架与 UI 界面的联系留作 TODO 内容供未来开发。

### 一、代码实现思路

首先我的代码框架包含 User、Task、TimeLine 等主要的类



User 类含有 Task 类的成员数组来管理创建的任务,同时还有 Notification 类的成员数组来实现消息提醒功能,Timeline 类成员用以展示任务时间线视图。

```
class User {
private:
    int userId; //用户ID
    string userName; //用户昵称
    vector<Task*> tasks; //已有任务列表
    vector<Notification*> userNotifications; //提醒消息列表
    Timeline Tline; //时间线视图

public:
    int taskCount; //任务序号
    int notificationCount; //消息序号
    User(int id, const string& name) : userId(id), userName(name),
        taskCount(0), notificationCount(0) {}

    void addTask(); //新建任务
    void editTask(int taskId); //编辑已有任务
    void deleteTask(int taskId); //删除任务
    void addNotification(Notification* n); //增加提醒消息
    void changeNotification(Task* t); //修改提醒消息
    void deleteNotification(int id); //删除提醒消息
    void listTasks() const; //显示任务界面
    void TimeList(); //显示任务时间线视图
    void listNotifications(); //发出提醒消息
};
```

当用户选择新建任务时,根据用户输入设置新任务的各种信息,同时将新任务加入任务列表和任务时间线列表中,然后再新建一个消息提醒并加入消息列表中:

```

void User::addTask() {
    Task* task = new Task(++taskCount); // 新建一个任务
    task->newTask(); // 设置任务各种信息
    tasks.push_back(task); // 将新任务加入任务列表中
    Tline.addTask(task); // 将新任务加入时间线视图中
    // 新建一个消息提醒
    Notification* n = new Notification(++notificationCount, task->taskId,
        task->description, task->deadline, task->reminderInterval);
    addNotification(n); // 将新的消息提醒加入消息列表中
}

```

修改任务和删除任务都先在任务列表中查找任务，然后修改任务各类信息后同步任务列表和任务时间线，并相应地修改消息提醒。

```

void User::deleteTask(int taskId) {
    int fflag = 0;
    // 查找任务
    for (auto iterator = tasks.begin(); iterator != tasks.end(); iterator++) {
        if ((*iterator)->taskId == taskId) {
            fflag = 1;
            break;
        }
    }
    if (fflag == 0) {
        cout << "没有这个任务哦！重新选择一下吧！\n";
        return;
    }
    // 删除任务
    tasks.erase(remove_if(tasks.begin(), tasks.end(),
        [taskId](Task* task) { return task->taskId == taskId; }),
        tasks.end());
    Tline.tasks.erase(remove_if(Tline.tasks.begin(), Tline.tasks.end(),
        [taskId](Task* task) { return task->taskId == taskId; }),
        Tline.tasks.end());
    deleteNotification(taskId);
}

```

而主函数则会先设置用户各种信息，然后不断判断用户的选择来执行相应的动作。

```

// 创建用户
User user(1, name);
int flag = 1;
int choice;
cout << "你想做什么呢？（输入选项前的数字我就可以帮你完成喔！）\n"
    << "0 退出程序， 1 查看任务单， 2 查看任务时间线页面， "
    << "3 新建任务， 4 修改任务， 5 删除任务\n";
cin >> choice;
if (choice == 0)
    flag = 0;
// 不断判断用户做出的选择
while (flag) {
    if (flag == 0) {
        cout << "要离开了吗？期待下次见面咯！\n";
    }
    return 0;
}

```

同时会不断检测消息提醒列表中的消息是否该被发出：

```

void User::listNotifications() {
    if (this->userNotifications.size() == 0) {
        return;
    }
    // 计算当前时间和DDL的间隔
    chrono::system_clock::time_point current_time = chrono::system_clock::now();
    vector<Notification*> UN = &userNotifications;
    for (auto iterator = (*UN).begin(); iterator != (*UN).end(); iterator++) {
        chrono::minutes interval = chrono::duration_cast<chrono::minutes>((*iterator)-
            // 判断是否需要发出消息
            if ((*iterator)->flag && interval.count() <= (*iterator)->reminderInterval) {
                // 发出通知
                (*iterator)->sendNotification();
                deleteNotification((*iterator)->taskId);
            }
    }
}

```

最后软件源代码的总规模在 655 行。

我利用大模型帮我根据 UML 类图生成了大致的代码框架，并对一些不太符合软件需求的部分进行了修改和补充，同时代码框架中一些不太理解的部分也通过大模型帮我解释来更快地理解。

## 二、代码的编译和运行结果

### ● 新建任务

```
hmmmm@Hmmm:/mnt/d/_MyCollege/Junior_1/SoftwareProject/实验三/code$ ./test-11-28
欢迎使用ToDoList, 给自己起个名字吧!
Your Name:Rose

你想做什么呢? (输入选项前的数字我就可以帮你完成喔!)
0 退出程序, 1 查看任务单, 2 查看任务时间线页面, 3 新建任务, 4 修改任务, 5 删除任务
3

请输入任务标题: Homework
请输入任务描述: Finish problems on the textbook!
请设置DDL: 年: 2024
月份: 12
日期: 30
小时: 12
分钟: 0
秒数: 0
请输入提醒间隔(min): 60
请设置任务类别 (输入选项前的数字)
0 工作, 1 生活, 2 娱乐, 3 其他
0
请设置任务标签颜色 (输入选项前的数字)
0 红色, 1 黄色, 2 绿色, 3 蓝色, 4 粉色, 5 黑色, 6 棕色, 7 紫色, 8 白色
1
请设置任务优先级 (输入选项前的数字)
0 高, 1 中, 2 低
0
接下来还想让我帮你做什么呢?
0 退出程序, 1 查看任务单, 2 查看任务时间线页面, 3 新建任务, 4 修改任务, 5 删除任务
```

### ● 查看任务单

```
接下来还想让我帮你做什么呢?
0 退出程序, 1 查看任务单, 2 查看任务时间线页面, 3 新建任务, 4 修改任务, 5 删除任务
1

*****
TaskID: 1
Task: Homework
Description: Finish problems on the textbook!
Deadline: Mon Dec 30 12:00:00 2024

Priority: High
Status: Unstarted
*****
TaskID: 2
Task: Basketball Game!
Description: Fight against TeamA Tue.
Deadline: Sun Dec 29 08:30:00 2024

Priority: Medium
Status: Unstarted
*****
```

- 查看任务时间线（根据 DDL 顺序排列，第二个任务 DDL 早于第一个任务，应该排在第一个任务前面，如图）

```

接下来还想让我帮你做什么呢？
0 退出程序， 1 查看任务单， 2 查看任务时间线页面， 3 新建任务， 4 修改任务， 5 删除任务
2

*****
TaskID: 2
Task: Basketball Game!
Description: Fight against TeamA Tue.
Deadline: Sun Dec 29 08:30:00 2024

Priority: Medium
Status: Unstarted
*****
TaskID: 1
Task: Homework
Description: Finish problems on the textbook!
Deadline: Mon Dec 30 12:00:00 2024

Priority: High
Status: Unstarted
*****

```

## ● 修改任务

```

接下来还想让我帮你做什么呢？
0 退出程序， 1 查看任务单， 2 查看任务时间线页面， 3 新建任务， 4 修改任务， 5 删除任务
4

告诉我你想修改哪个任务呢？（输入0可以退出）
1

请选择修改的内容（输入选项前的数字）：
0 退出， 1 任务标题， 2 任务详情， 3 截止时间， 4 提醒间隔， 5 类别， 6 优先级， 7 状态
7

请设置任务状态（输入选项前的数字）
0 已完成， 1 正在进行， 2 即将到期， 3 未开始
1

请选择修改的内容（输入选项前的数字）：
0 退出， 1 任务标题， 2 任务详情， 3 截止时间， 4 提醒间隔， 5 类别， 6 优先级， 7 状态
3

请设置DDL：年：2024
月份：12
日期：29
小时：6
分钟：30
秒数：0

```

此时第一个任务的 DDL 早于第二个任务，时间线视图中应该在第二个任务之前，如图：

```

接下来还想让我帮你做什么呢？
0 退出程序， 1 查看任务单， 2 查看任务时间线页面， 3 新建任务， 4 修改任务， 5 删除任务
2

*****
TaskID: 1
Task: Homework
Description: Finish problems on the textbook!
Deadline: Sun Dec 29 06:30:00 2024

Priority: High
Status: Underway
*****
TaskID: 2
Task: Basketball Game!
Description: Fight against TeamA Tue.
Deadline: Sun Dec 29 08:30:00 2024

Priority: Medium
Status: Unstarted
*****

```

## ● 删除任务

```

接下来还想让我帮你做什么呢？
0 退出程序， 1 查看任务单， 2 查看任务时间线页面， 3 新建任务， 4 修改任务， 5 删除任务
5

告诉我你想删除哪个任务呢？（输入0可以退出）
1

接下来还想让我帮你做什么呢？
0 退出程序， 1 查看任务单， 2 查看任务时间线页面， 3 新建任务， 4 修改任务， 5 删除任务
1

*****
TaskID: 2
Task: Basketball Game!
Description: Fight against TeamA Tue.
Deadline: Sun Dec 29 08:30:00 2024

Priority: Medium
Status: Unstarted
*****

```

时间线视图中该任务也被删除

```

接下来还想让我帮你做什么呢？
0 退出程序， 1 查看任务单， 2 查看任务时间线页面， 3 新建任务， 4 修改任务， 5 删除任务
2

*****
TaskID: 2
Task: Basketball Game!
Description: Fight against TeamA Tue.
Deadline: Sun Dec 29 08:30:00 2024

Priority: Medium
Status: Unstarted
*****

```

- 当某任务到达提醒时间时会发出提醒：

```

*****
Notification: 2
TaskId: 2
Message: Fight against TeamA Tue.
The task is to due, remind to finish it!
*****
接下来还想让我帮你做什么呢？
0 退出程序， 1 查看任务单， 2 查看任务时间线页面， 3 新建任务， 4 修改任务， 5 删除任务

```

### 三、GIT 远程代码管理展示

-