



程序设计实习——QT 项目

项 目 名 称:

队 名: 未名队

成 员: 刘晋铭 魏景丽 马知微

二〇二三年 七月

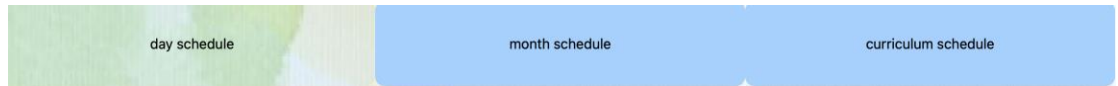
一、程序功能介绍

我组的项目是基于 QT 的日程管理器。主要功能分为以下几点：

1. Day Schedule
对待办事项进行添加、删除、修改。
对每日主题（重要事项）进行添加、删除和修改。
用户可以在右侧的按钮中选择任意日期，之后跳转到对应的日期界面。
2. Month Schedule
切换不同日期界面可以显示不同日期下的日程安排。
用户可在选中的日期的界面添加当日的日程主题。
可以对日程安排进行相应的修改与保存。
3. Curriculum Schedule
添加对应的课程，以及课程的相关信息
同时用户可以设置对应课程的主题颜色。
4. 在界面打开时，会有背景音乐循环播放

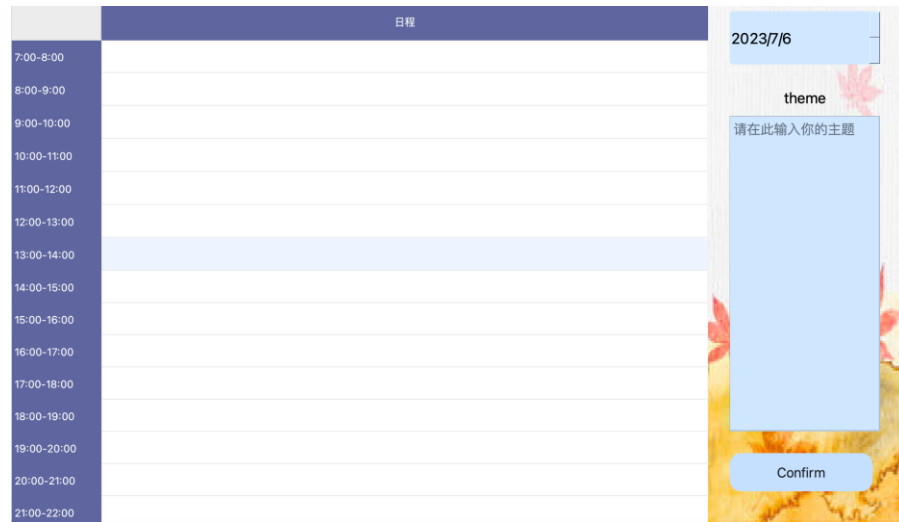
二、项目模块与类设计细节

1. 后端部分
 - i. Day Schedule 模块
先实现了一个 task 类, 其中包含起始时间, 内容, 是否完成信息; 然后实现了 daylist 类, 包含一个 task 类的 QVector, 当日主题, 当日日程, 当日日期信息; MainWindow 类里包含一个 daylist 的 QVector, 称为 bank_daylist。通过前端在 Day Schedule 界面更改日期和当前日期下的某一时间段任务内容, bank_daylist 中的内容随之更新, 有当日日期和当前日期相同的 daylist 则修改或添加某一时间段的任务内容（即更改其中 task 类的 QVector），无则添加新的 daylist。
 - ii. Month Schedule
与 Day Schedule 模块基本类似，更新的时候更新的是 daylist 中的 dayschedule。
 - iii. Curriculum Schedule
在 MainWindow 类里维护两个 QString 字符串数组和一个 int 数组，分别记录上课名字、上课地点和主题颜色，数组都是二维的，第一个维度代表第几节课（从 0 开始），第二个维度代表星期几（也是从 0 开始），通过前端在 addcurriculum 界面修改课程表内容更新三个数组对应的内容。
2. 界面部分
主界面采用 tab widget 显示三个对应的功能区，并添加了背景图片以美化界面



i. Day Schedule 模块

主体采用 table widget 的表格形式，显示对应的时间与相应时间的日程。
右侧区域用 dateedit 来供用户选择对应的日期。

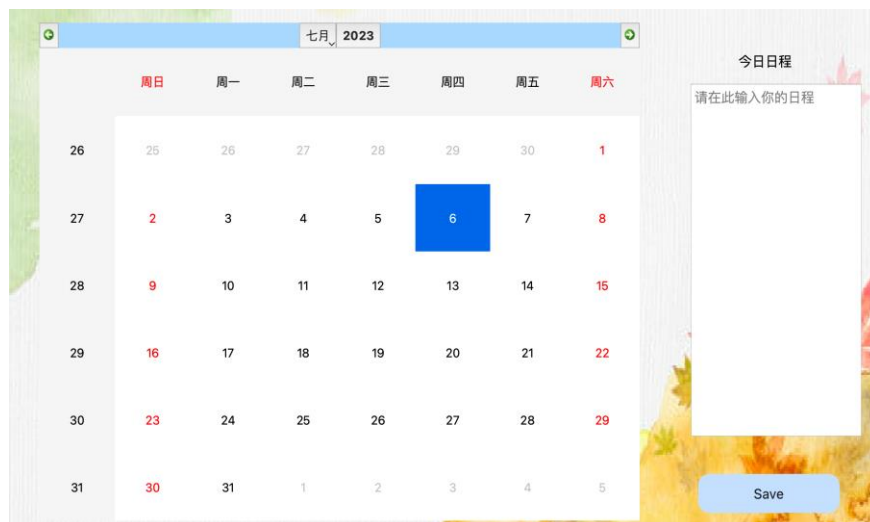


下方有 textedit，并设置了对应的 PlaceholderText（如下图所示：“请在此输入你的主题”）用以提示用户的输入，用户可输入对应的日程主题。
输入完成之后可以点击 confirm（pushbutton），之后会弹出对话框，用户确认是否进行保存。



ii. Month Schedule 模块

在本模块借助了 QT 自带的控件 CalendarWidget，右侧同样采用相应的 textedit，并设置了对应的 PlaceholderText。
设置完成之后，可以点击下方的 pushbutton，弹出对话框后可以确认是否保存。

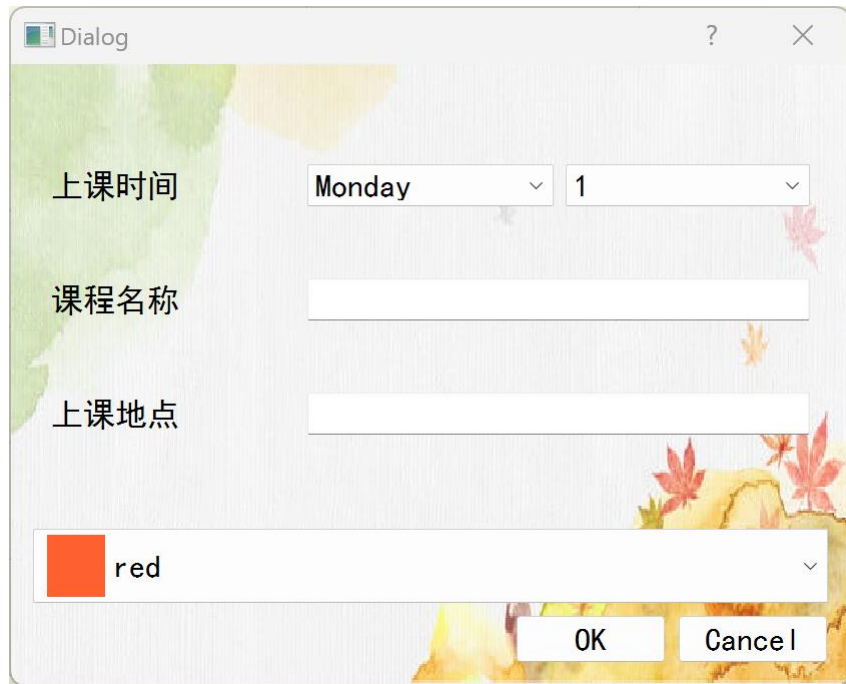


iii. Curriculum Schedule 模块

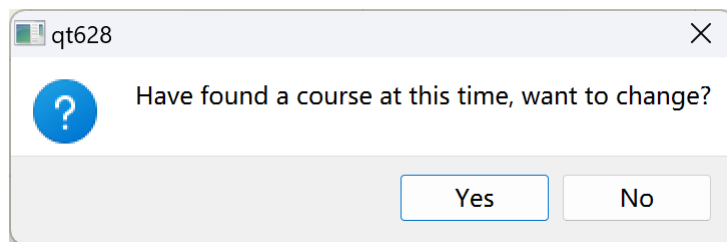
本模块中采用了 TableWidget，点击右上角的加号，会弹出相应的对话框
 可以在此进行课程时间、课程名称、上课地点、主题颜色的设置。
 为了直观的区分不同课程，我们在此提供了设置课程主题颜色的功能。

	Mon	Tue	Wed	Thur	Fri	Sat	Sun
1	程设 二教407						+
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							

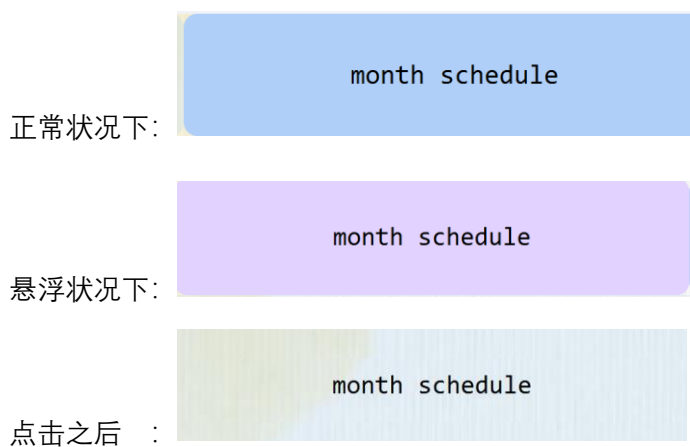
对话框总体采用栅格布局与垂直布局相结合的方式，这样可以保证内部控件的大小与相对位置不受对话框窗口大小的影响。



如果用户在已有课程的时间段内添加新的课程，会弹出对话框询问用户是否进行覆盖。



- iv. 总体的界面设计，我们采用了统一的色调，同时借助样式表，我们将原有的 qt 控件的风格进行了很大程度的美化与润色，提升了用户的视觉体验。同时 bgm 的添加，也为用户提供一个轻松愉悦的工作氛围。
- v. 在对应的按钮设计中，我们精心把握了按钮的颜色，例如按下，悬浮，正常状况下的按钮状态有所区分



三、小组成员分工情况

1. 刘晋铭：后端部分的实现，部分实验报告的撰写

2. 魏景丽: day schedule, month schedule 部件的设计与美化, 部分实验报告的撰写, 展示视频的录制
3. 马知微: curriculum schedule 界面和部件的设计与美化, 部分实验报告的撰写, 展示视频的录制

四、项目总结与反思

1. 项目整体代码量不算大, 但是 debug 时间比较长, 经过总结, 发现大多数 bug 来源于将 QDateTime 对象转换为字符串时, 调用的 toString 函数里面没有指定转换格式而是直接写 xxx.toString(), 反之将字符串转换为 QDateTime 对象时调用 fromString 函数也有类似的错误。
2. 由于是刚刚使用 QT, 大部分内容是自学的, 代码量不大的项目写起来仍然比较费时间, QT 中的帮助选项可以给予用户很多各个 QT 自带类的函数名、函数功能等方面的提示, 有效地解决了初学者使用 QT 的部分障碍。同时, 自学 QT 也极大地锻炼了我们上网查找信息的能力, 通过在 b 站、csdn 等网站上查找相应的网课和博客等, 我们得到了许多关于界面设计与美化的灵感。
3. 项目最终实现的时候并没有完全按照当初的功能设计文档走, 省略的功能中部分是因为技术上的原因, 添加的功能中部分是出于对用户体验感因素的考量。
4. 最终版本与初稿的界面形式做了很大的改动, 之前的版本中, 界面比较杂乱, 部件也存在冗余现象, 日界面和月界面采用两个 mainwindow 来实现, 在美观程度和用户使用体验上都有所不足。因此, 我们在期末考试后重新设计了现有界面, 删除了冗余的对话框和功能, 尽力为用户提供一个简洁直观, 同时得心应手的日程管理器。但是, 由于技术和时间的限制, 我们仍有一些功能未能及时完成和完善。
5. 通过一起做本次项目, 我们也对团队合作有了更多更深的感悟, 同时, 我们的沟通、协作能力也在前后端人员间的反复交流、配合中得到了很大的提高。