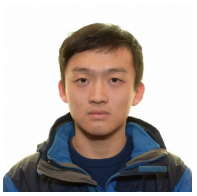
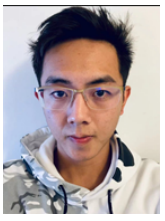
Interface Design for a Meeting Scheduling Tool

Team name: C++ Addictor

Date: Oct 2nd, 2021

Team members:

Te Lai laite19991205@gmail.com		Lezhi Wang wanglz2000@outlook.com	
Sheng Wang wangshengwalter@gmail.com		Zishu Xia leonxia1018@gmail.com	

Section 1: problem definition

The main task of this application is to schedule meetings for a group of users. The functionality which need to be achieved includes:

1. Collecting the schedules from all members and displaying them in a combined timetable.
2. Receiving the meeting information (location, duration, topic, etc.) from the meeting holder.
3. Providing choices of time when most people are available based on schedules which all attendees submit for the meeting holder.
4. Receiving the time chosen by the meeting holder.
5. Informing all attendees of the meeting information and asking if they are able to attend.
6. Informing the meet holder of the estimated attendance.
7. Being able to inform all attendees right before the meeting.

Two groups of users are major target users. First, some small teams such as white-collar workers and students have few opportunities to meet face to face to decide when to meet during the pandemic, and the software builds an online platform for them to schedule a meeting online. Another group of users are leaders of large organizations, such as company managers and government officials. They can organize a large number of people to find an optimal time period in a short period of time by using this interface.

The level of support (sorted by priority):

1. The interface provides a large visible timetable which is easy for users to identify.
2. The interface has no more than 10 buttons with stringent forward hints.
3. Users can learn to use this software within two minutes without using the same type of software before.
4. The meeting holders are able to reorganize the meeting within 4 steps.
5. 95% of target users are satisfied with this interface.
6. 100% of the interfaces have a back button to return to the previous interface and automatically save the current content entered by the user. All uploaded content should upload successfully after users click the confirm button to avoid misclicking.
7. The time of uploading and displaying a member's work schedule on the combined timetable is 10ms.
8. The error rate of misclicking is lower than 0.1%.

The form of solution:
An online application that includes a scheduler tool for time management and coordinating meetings.

Section 2: analysis

After looking through most of the conference/meeting schedulers online, we find that the meeting scheduler in Outlook is the relatively most complete one. The scheduler in Outlook contains lots of useful functionalities for its target users, but still remains some unintuitive interaction issues to be improved.

As a time scheduler that is widely used by various organizations, Outlook scheduler does have some positive aspects:

1. The scheduler allows the meeting holder to book a meeting recurrently. (e.g. weekly meetings, monthly meetings). (Figure 1)
2. The scheduler allows the holder to book a physical meeting room. Physical rooms are virtualized to be email addresses and the availability of the rooms are handled by the remote server. Redundant bookings will be rejected and holders will be informed by email. (Figure 2 & Figure 3 & Figure 4)
3. After the holder adds all attendees, a "bar table" will be auto-generated by the system. The bar table allows the holder to check all attendees' timetables and choose a time slot that everyone is available by clicking and dragging. This functionality offers lots of useful information and is easy to comprehend. (Figure 5)
4. The system will remind every attendee who accepts the meeting 15 minutes (default but configurable) in advance. (Figure 6)

On the other hand, Outlook has some interaction issues that really confuse users, especially meeting holders:

1. The UI of meeting booking is relatively messy and makes the users confused. Too many input boxes are on the same page, with a big blank space remaining on the page, which wastes the space of the page. (Figure 7)
2. Instead of entering the information required step-by-step, the user needs to enter all of the information one time, which makes the whole process confusing and error-prone.
3. If there are conflicts in meeting time or places, users are still allowed to send meeting requests to the server, which is meaningless and dramatically increases the latency of the process. In this case, the server will reject the request after a while. Besides, users are very likely to ignore the delayed responses, and not realize that the request was rejected.

Section 3: suggested improvements

Having figured out some drawbacks of the Outlook interface, we come up with some improvement ideas. We focus on user experience as well as efficiency of the workflow, and our suggestions are listed below:

1. The booking process should be in series instead of parallel. In current Outlook, users are required to enter all information (holder's name, attendees' name, time, location, description, reminder, repeat or not, etc.) of a meeting one time. Our suggestion is that the booking process is supposed to be divided in a few separate steps, for example, the holder should first select the attendees, then click "Next", then select time and location. People are likely to feel confused and anxious when they are facing many input boxes at the same time. Also, they are more likely to make mistakes under such circumstances.
2. Redundant room booking requests should never be sent. During the booking process, the system should be able to identify whether the rooms proposed in the request are available or not. If not, the system should be able to notify the holder that the room is unavailable and ask the holder to adjust the location or time. As shown in Figure 8 and 9, the room handlers should not be able to respond to the holder. Instead, it is only asked to book the time slot. In this way, unavailable meetings will not exist and people will not come to unavailable rooms to attend meetings. Holders will be able to organize meetings with no location conflicts and not receive emails saying "Your

booking request is denied due to conflicts", delete it and send another request any more, which saves their time.

3. Many pages in Outlook should be intuitive enough. Some vital components such as scheduler assistant are hidden deeply and hard to find. In the meeting bookings UI shown in Figure 6, there are too many input boxes listed with much space still blank, which makes it look messy. We believe that a UI designer should make full use of space and try to keep a reasonable amount of elements in a single page. For instance, we can make the meeting booking process a multi-step process instead of a one-step process, which means we can increase the number of pages in order to reduce the number of elements on a single page.

The Flow chart of the Outlook logic and the logic we intended are attached in Appendix (Figure 8 & 9).

Reference

[1] Booking a meeting room in Outlook. (2017, November 16). Www.youtube.com. Retrieved October 2, 2021, from https://www.youtube.com/watch?v=sMcoS-JVzEo

[2] Scheduling Meeting Rooms in Microsoft Outlook. (2015, March 27). Www.youtube.com. Retrieved October 2, 2021, from https://www.youtube.com/watch?v=SasZL15buIU.

Appendix

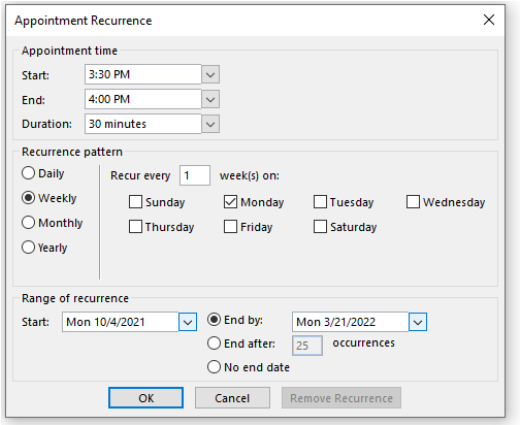


Figure 1. Meeting Recurrence Configuration

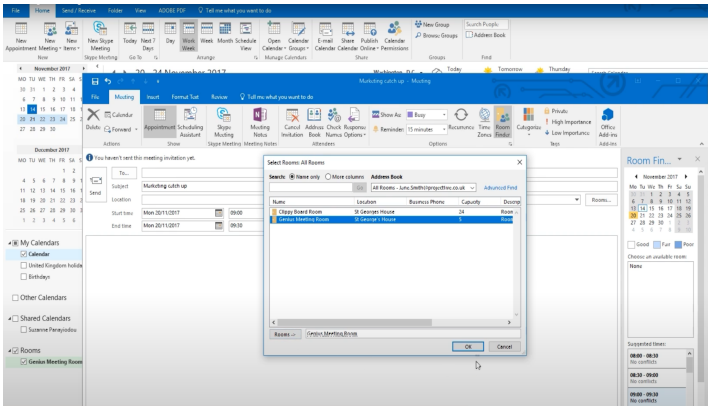


Figure 2. Choose the Meeting Room

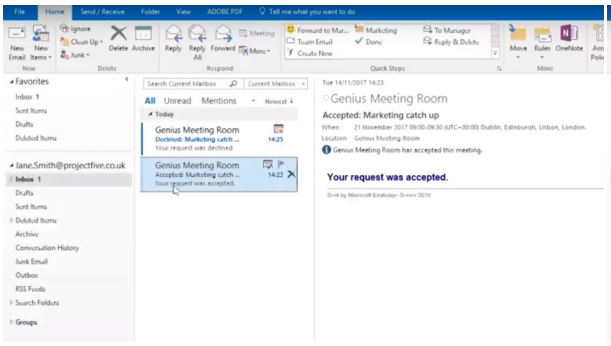


Figure 3. Room Booking was accepted[1]

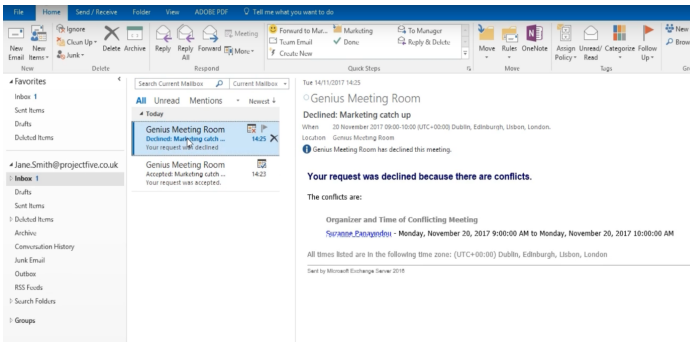


Figure 4. Room Booking was rejected[1]

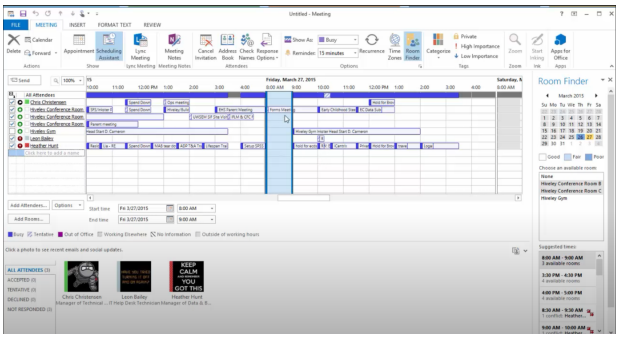


Figure 5. Room Booking was rejected[2]

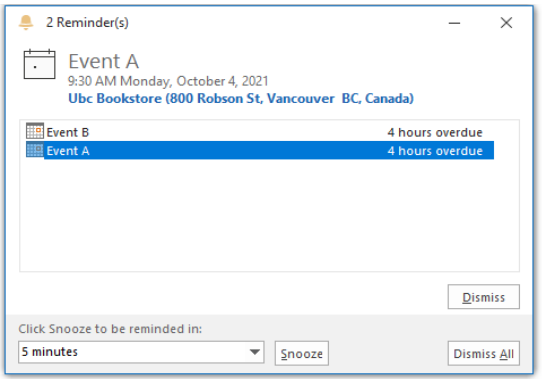


Figure 6. Meeting Reminder

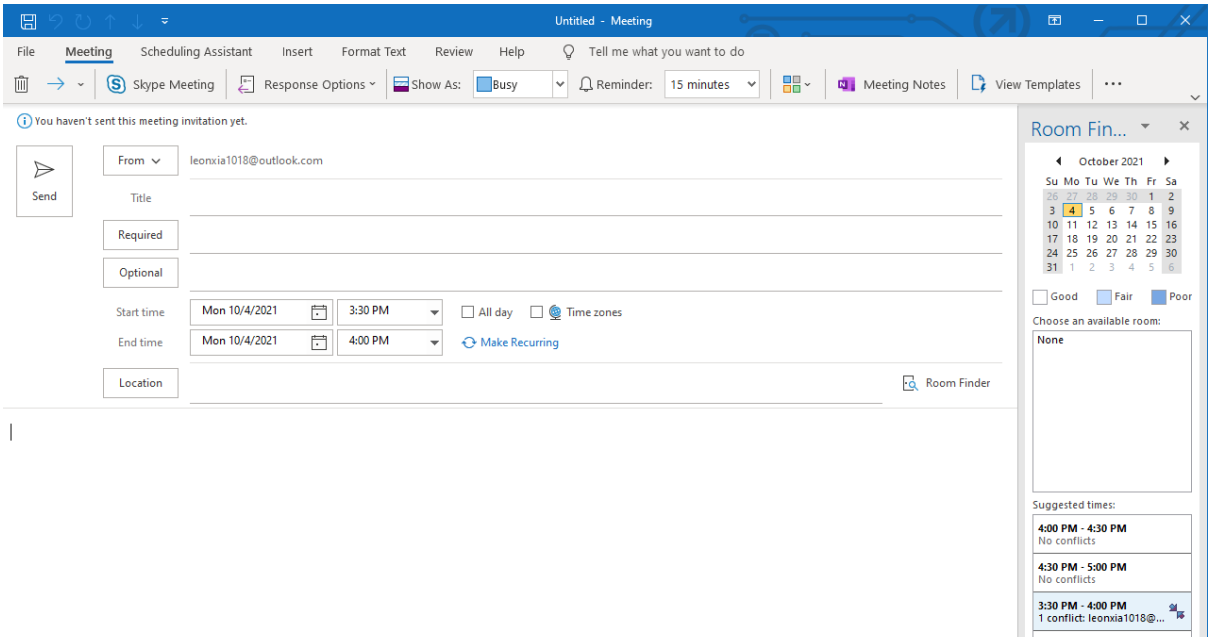


Figure 7. Meeting Booking UI

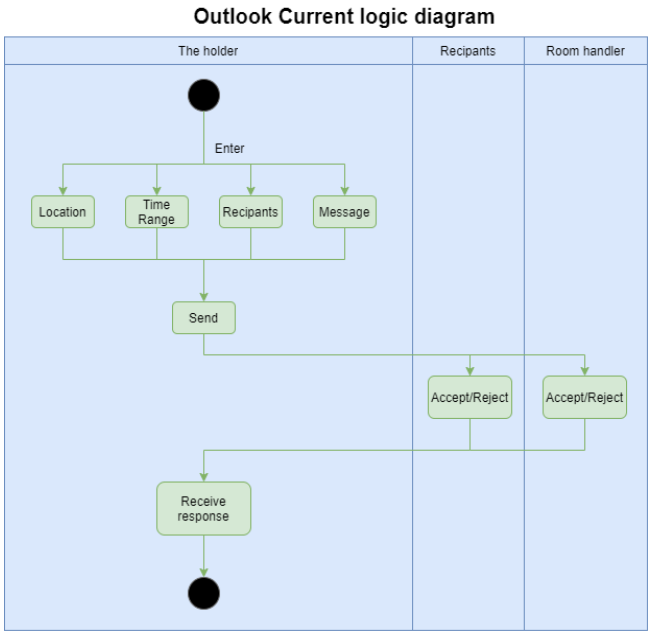


Figure 8.

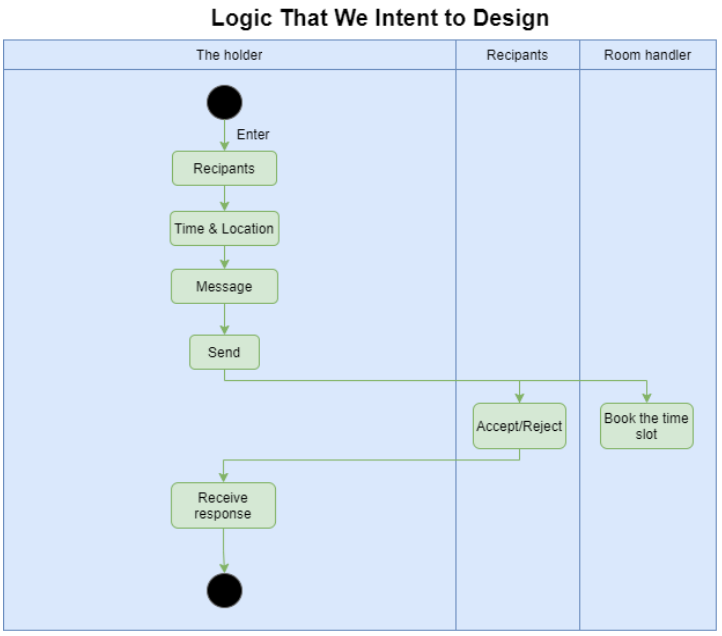


Figure 9.