Task 1 Individual Project

This document contains the specification for the individual homework assignment (Due noon September 25th). **New requirements are in red.**

The organizer wishes to set up a calendar of timeslots. Each timeslot has a specific date and start time. From the timeslot date you can determine the day of week to which it belongs (which must be Mon, Tue, Wed, Thu or Fri).

All meetings in the calendar have the same duration. A calendar has a default location for all meetings which can be overridden, as needed, for any given meeting.

Use Cases

1. Create calendar (each calendar has a name (i.e., private, professional) with specific starting date and ending date, earliest hour, latest hour, and duration of each slot (either 10 min, 15 min, 20 min, 30 min, 60 min).
   Note: Once timeslots are set up they are enforced for the entirety of the calendar; that is, each meeting is always going to be that same duration. The only way to change the duration is to delete entire calendar and start again.
2. Delete calendar (by name)
3. Load calendar (by name)
4. Add day to calendar, which extends the potential timeslots available for meetings
5. Remove day from calendar, which removes timeslots
6. Close timeslot so it can't be selected as a potential meeting. To make this user friendly, the organizer can close (a) an individual timeslot (i.e., Sep-05-2018 @ 11:00 AM); (b) all timeslots on a given Day of Week and time (i.e., Tuesdays @ 2:00 PM); (c) all timeslots on a given Day (i.e., All meetings on Sep-7-2018); **(d) all timeslots at a given time (i.e., 3PM on any day)**
7. Schedule meeting (i.e., date & time) with specific person and an optional new location
8. Cancel meeting (i.e., date & time)
9. Show daily schedule for a given date
10. Show monthly schedule
11. Double-check to make sure I haven't missed anything…

StoryBoards

Mock up some sample GUIs describing how the organizer will interact with the system. Use your notebooks. Try to envision each step of the user experience from the point of view of the organizer.

Concrete Scenarios

To receive full credit for this assignment, you must demonstrate your application completes the following steps. Your GUI elements you design will have to support the following behaviors. Be sure in your final assignment to provide sufficient explanation to me so I can carry out these steps without wondering how to do them.

The name of the system you develop is called "CMS" for Calendar Management System. Since all of the steps are carried out from the perspective of the Organizer, they are briefly outlined. Thus instead of saying "Organizer executes CMS" I just say "Execute CMS"

Create Personal Calendar [20 pts]

1. Execute CMS
2. Create a new calendar "Personal" which will cover the dates from Monday September 24th 2018 through Friday October 5th 2018. All meetings will have a 20 minute duration, and the first timeslot will start at 10AM on each day (i.e., 10:00a – 10:20a), and the last timeslot will end at 5PM (i.e., 4:40p – 5:00p).
3. Exit CMS

Load Personal Calendar and Create First Meeting [20 pts]

1. Execute CMS
2. Request to load calendar "Personal"
3. Observe that Personal calendar exists and organizer can see the available dates for that calendar
4. Request to schedule a meeting for Tuesday September 25th 2018 from 1:20p – 1:40p with "George Heineman"
5. Observe that meeting has now been claimed for that timeslot.
6. Request to schedule a meeting for Wednesday September 26th 2018 from 10:40a – 11:00a with "Mickey Mouse"
7. Observe that second meeting has now been claimed for that timeslot.
8. Exit CMS
9. Execute CMS
10. Load "Personal" calendar
11. Confirm the there is a meeting on Tue Sep 25th at 1:20 with "George Heineman" and a second meeting on Wed Sep 26th at 10:40a with "Mickey Mouse".
12. Exit CMS

Add Day to Calendar [10 pts]

1. Execute CMS
2. Request to load calendar "Personal"
3. Observe that Personal calendar exists and organizer can see the available dates for that calendar
4. Add date Friday October 12th to the calendar
5. Observe that calendar now runs from September 24th through October 12th
6. Request to schedule a meeting for Friday October 12th 2018 from 2:00p – 2:20p with "Buzz Lightyear"
7. Observe that third meeting has now been claimed for that timeslot
8. Exit CMS

Remove Day from Calendar [10 pts]

1. Execute CMS
2. Request to load calendar "Personal"
3. Observe that Personal calendar exists and organizer can see the available dates for that calendar
4. Remove date Tuesday October 2nd from the calendar
5. Attempt to schedule a meeting for Tuesday October 2nd (say at 10:00a – 10:20a) but demonstrate it can't be done.
6. Exit CMS

Show Daily Schedule [10pts]

1. Execute CMS
2. Request to load calendar “Personal”
3. Observe that Personal calendar exists and organizer can see the available dates for that calendar
4. Request to show daily schedule for September 26th 2018 and confirm there is a meeting for that day with “Mickey Mouse”
5. Exit CMS

Show Monthly Schedule [5 pts]

1. Execute CMS
2. Request to load calendar “Personal”
3. Observe that Personal calendar exists and organizer can see the available dates for that calendar
4. Request to show monthly schedule for September 2018 and confirm there are two meetings, one with “George Heineman” (Sep-25-18 @ 1:20p) and “Mickey Mouse” (Sep-26-18 @ 10:40a)
5. Exit CMS

Close Timeslots [5 pts]

1. Execute CMS
2. Request to load calendar “Personal”
3. Observe that Personal calendar exists and organizer can see the available dates for that calendar
4. Request to close all timeslots at 3:00p (note: this is revised requirement omitted from last week)
5. Attempt to schedule a meeting for Monday October 1st (say at 3:00p – 3:20p) but demonstrate it can’t be done.
6. Exit CMS

Close Timeslot and observe doesn’t affect existing meetings [5 pts]

1. Execute CMS
2. Request to load calendar “Personal”
3. Observe that Personal calendar exists and organizer can see the available dates for that calendar
4. Request to schedule a meeting for Friday September 28th 2018 from 4:00p – 4:20p with “Donald Duck”
5. Request to close all timeslots on Friday September 28th
6. Show daily schedule for September 28th 2018 and confirm there is a meeting for that day with “Donald Duck”
7. Attempt to schedule a meeting for Friday September 28th (say at 4:20p – 4:40p) but demonstrate it can’t be done.
8. Exit CMS

Cancel Existing Meeting [10 pts]

1. Execute CMS
2. Request to load calendar “Personal”
3. Observe that Personal calendar exists and organizer can see the available dates for that calendar
4. Request to cancel meeting for Tuesday September 25th 2018 from 1:20p – 1:40p with “George Heineman”

5. Show daily schedule for September 25th 2018 and confirm there is no meeting with “George Heineman”
6. Exit CMS

Delete Calendar [5 pts]

1. Execute CMS
2. Create a new Calendar “Professional” which will cover the dates from Tuesday October 2nd 2018 through Tuesday October 2nd 2018. All meetings will have a 30 minute duration, and the first timeslot will start at 10AM on each day (i.e., 10:00a – 10:30a), and the last timeslot will end at 5PM (i.e., 4:30p – 5:00p).
3. Exit CMS
4. Execute CMS
5. Request to load “Professional” and confirm it exists
6. Request to delete “Professional” calendar
7. Exit CMS
8. Execute CMS
9. Request to load “Professional” calendar and observe that it cannot be done since calendar was deleted
10. Exit CMS

End Notes

For convenience, you should consider the following implementation decisions.

1. Read the calendar into memory at the start of your program and write the whole thing out to disk when you exit the program. Note: this is not an ideal strategy as you will see for Task 2, but I’m trying to simplify your implementation without overly complicating your design.
2. Persistently store “the last calendar opened” so when you execute CMS, it automatically loads up the most recently opened calendar. Feel free to store in formation in the user’s home directory, i.e., `System.getProperty("user.home")`