Unit:
Analysis, Design and Implementation

Assignment title:
Meeting Scheduler

December 2018

Important notes

- Please refer to the Assignment Presentation Requirements for advice on how to set out your assignment. These can be found on the NCC Education *Campus*. Click on Policies and Advice in the left-hand menu and look under the Advice section.
- You must read the NCC Education documents 'What is Academic Misconduct? Guidance for Candidates' and 'Avoiding Plagiarism and Collusion: Guidance for Candidates' and ensure that you acknowledge all the sources that you use in your work. These documents are available on *Campus*. Click on Policies and Advice in the left-hand menu and look under the Policies section.
- You **must** complete the **'Statement and Confirmation of Own Work'**. The form is available on *Campus*. Click on Policies and Advice in the left-hand menu and look under the Policies section.
- Please make a note of the recommended word count. You could lose marks if you write 10% more or less than this.
- You must submit a paper copy and digital copy (on disk or similarly acceptable medium). Media containing viruses, or media that cannot be run directly, will result in a fail grade being awarded for this assessment.
- All electronic media will be checked for plagiarism.

Scenario

Meetings are one of the key things that drive forward any organisation, but they can be very difficult to schedule. Everyone has other things they should be doing, and the presence of some people is more important than others. Some meetings require a minimum attendance (sometimes known as **quorum)**, some just need a handful of the right people. Your task for this assessment is going to be to develop a meeting scheduler that can be used to make sure that the right numbers of people, of the right type, are invited to the right meetings.

The system will track individual users, who will be permitted access to a calendar interface that they will fill with their professional obligations. For this, they will provide a day, time and duration as well as whether their presence is vital or optional. Any time that is not assigned is defined as 'free time' that is available for allocation to meetings. Time that is spent on an activity where their presence is 'optional' can be used to make up numbers or ensure vital people are at important meetings.

For each user in the system it should track:

1. Name
2. Job Title
3. Pay Grade
4. Gender
5. Age
6. Calendar

The system administrator can create **meetings**, and these have the following key pieces of information that need to be set:

1. Date
2. Time
3. Duration
4. Minimum attendance
5. Maximum attendance
6. Description
7. Specific attendees
8. Declined invitations
9. Accepted invitations
10. Tentative invitations

The system should, upon being given specific attendees, be able to generate a suggested time when all of the people involved are either available or engaged in activities where their presence is optional. The system should then suggest attendees based on the minimum required attendance and the availability of people at the suggested time slots. When a time slot has been set, it should be automatically recorded in each attendee's calendar. Users should be presented with a list of meetings to which they have been allocated and permitted to accept, decline or issue a tentative acceptance. If someone is a specifically requested attendee, they are automatically deemed to have accepted. When someone declines an invitation a new invitation should be generated and sent out to someone not previously invited.

Your application then needs to provide the following functionality:

- Allows for users to register themselves with the system and edit their details.
- Allows for users to update their calendars.
- Allows for new meetings to be added and edited
- Allows for specific attendees to be invited to a meeting and for suggested time-slots to be presented.
- Sends invites out to users in the system based on the minimum required for being quorate.
- Sends invites to users and permits them to accept or decline.
- Send new invitations out based on declined invitations.
- Allows users to view their calendar and the system administrator to view all the scheduled meetings.
- Saves the calendar and user data in an appropriate format.

Your solution will consist of a class diagram, a use-case diagram, an activity diagram for the process of finding a suitable time slot for specific attendees. You should also submit the completed program code in an appropriate programming language.

Task 1 – 26 Marks

Candidate class list and Diagrams

26 Marks are available for providing an appropriate list of candidate classes, along with the supporting diagrams. The candidate class list should incorporate justifications and discussion as to why each class was selected for inclusion, and how its relationship to other classes was derived. The class diagram should show attributes, operations, scope and relationship of classes to each other.

The marks for the task are broken down as follows: (1) 10 marks for Candidate class list; (2) 6 marks for Additional classes; and (3) 10 marks for Class diagram.

Task 2 – 25 Marks

Activity diagram

25 Marks are available for the creation of the appropriate activity diagram. The activity diagram should incorporate the classes involved in finding a suitable time slot for specified attendees.

The marks for the task are broken down as follows: (1) 20 marks for functionality; and (2) 5 marks for Class ownership.

Task 3 – 8 Marks

Use case diagrams

8 Marks are available for the provision of suitable use-case diagrams. The use case diagrams should incorporate each of the user activities indicated in the brief.

Task 4 – 15 Marks

Code architecture

15 Marks are available for a code architecture that shows an appropriate level of coupling and cohesion, along with the necessary amount of inheritance and encapsulation to express the system.

The marks for the task are broke down as follows: (1) 5 marks for handling the requirements via inheritance and polymorphism; (2) 5 marks for handling user input; and (3) 5 marks for handling output.

Task 5 – 26 Marks

System implementation

26 Marks are available for implementing the system as described and providing the completed C# code.

Submission requirements

- Your program must be submitted as a zip file of the full project.
    - Whatever IDE you use, it should be possible to open and run the project directly from the extracted archive.
- Diagrams and materials associated with the tasks above should be presented in a word-processed document.
- All references and citations must use the Harvard Style.

Candidate checklist

Please use the following checklist to ensure that your work is ready for submission.

Have you read the NCC Education documents 'What is Academic Misconduct? Guidance for Candidates' and 'Avoiding Plagiarism and Collusion: Guidance for Candidates' and ensured that you have acknowledged all the sources that you have used in your work?	❑
Have you completed the 'Statement and Confirmation of Own Work' form and attached it to your assignment? **You must do this.**	❑
Have you ensured that your work has not gone over or under the recommended word count by more than 10%?	❑
Have you ensured that your work does not contain viruses and can be run directly?	❑