OOD - Individual assignment: Workshop Management

You are tasked to create a Windows form application for the planning and management of workshops, for instance in an educational setting. People can either give a workshop or attend a workshop given by someone else. There are currently two types of workshops: 'online' and 'in building'.

Your application should be able to create and display workshops and assign people to them as either a teacher or a student. In addition, it should be possible to save an overview of planned workshops to a file and inform participants when a workshop is started.

Disclaimer

This is an individual assignment to give you better insight about your proficiency related to the covered concepts in OOD and the learning outcomes. Your teacher will give you a formative indication and feedback to help you understand your proficiency.

Note that this assignment is individual and is used by your teacher to help you get to the expected proficiency for OOD. For this reason, it must be your own code and you are not allowed to copy any part of it from your fellow peers. Your teacher will give you feedback about what you submit, and we assume that it is your own work and that you understand it fully.

Context:

You will be creating an application for an educational organization, to assist in the management and coordination of workshops.

Just like at Fontys, people within this organization are uniquely identified by a PCN number. Everyone in the organization can attend workshops as a student, but only people that are qualified by the organization to teach, can also present workshops as a teacher.

The organization currently employs a form of blended learning, meaning that workshops can either take place online or in a building. In addition to a title, there is also a short description of the topic stored for each workshop, as well as the person teaching it. Each workshop, no matter if it takes place online or in a building, also has a maximum capacity for participants. The main difference between online and in building workshops is the location: for in building workshops, the address and room number where the workshop takes place are stored, online workshops only have a URL where participants can join. In the future, different kinds of workshops may be added (such as a radio broadcasted workshop).

The end-user of this application is the receptionist at the organisation. They will use the application to perform actions on behalf of people that want to create and/or attend workshops, they must at least be able to do the following:

Functional requirements:

- **FR1: CRUD workshops**
  Workshops can be created in accordance with the description and added to the list of available workshops.
- **FR2: Add and modify a person**
  A person can be added to the system, registering their name and other relevant information.

- **FR3: Enrol a student**
  A person can be enrolled to a particular workshop as a student if there is a free spot.
- **FR4: Show workshop details**
  It should be possible to display the available information about a specific workshop, including an overview of the enrolled students.
- **FR5: Show enrolment details**
  It must be possible to view the enrolment details for a specific student in a workshop. In the case of an in building workshop, this should include an auto-generated seat number (indicating where the student should sit). For online workshops, the details should include an auto-generated login code.
- **FR6: Start a workshop**
  The user can start a workshop. This marks the workshop as started and locks it so editing and enrolment are no longer possible.
- ***FR7: Generate printable text file**
  It must be possible to save an overview of available workshops to a text file, so it can easily be printed. The format of the text file should look like this:
  ONLINE(Title:…, Description:…, Teacher:…, Capacity:…, URL:...)
  INBUILDING(Title:…, Description:…, Teacher:…, Capacity:…, Location: <address>
  <roomnr.>)
- ***FR8: Display public information separately**
  An overview of workshops that are available for enrolment should be shown in a separate display that could be shown to potential participants. If the current enrolment for a workshop exceeds 90% of its capacity, a warning should be shown that the workshop is almost full. (we expect you to implement this as a separate form and use event mechanics to trigger the warning)

Important information:

- Only people registered as a teacher can present workshops. If the user attempts to assign an unqualified person as teacher to a workshop, this should result in a (custom) exception being raised.
- Throughout your code, proper exception handling is expected where appropriate.
- You are expected to provide full unit tests for the class(es) that represent(s) the workshop.
- It is expected that you use the programming concepts covered during semester 1 and OOD correctly.
- For a high mark, consider adding extra features to your solution that contribute to the quality of the application. If you do, make sure to incorporate them in your design as well as your code.

The assignment

This assignment is split in two phases, in the first phase you analyse the case and design a class diagram. The deadline will be provided by your teacher in Canvas (before week 12). You will receive feedback from your teacher on class level (which classes you have and how they relate to each other) at the beginning of week 12.

In the second phase you will implement your design. Before the final deadline, you must submit both the final design and the finished application. Note that you do not need to wait for feedback to get started on phase two. Just make sure to review your design after and take any feedback into account.

Be sure to submit the solution of this assignment __before__ the deadline on Canvas; the final design as well as the implemented functionalities and code quality will be reviewed.