

## Software Engineering (IT): Assessed Exercise 2

## 软件工程 (IT) : 评估练习2

### **DEADLINE: Friday 17th March 2023; 17:00**

This is a follow up to Assessed Exercise 1 and, where possible, we will be using the same teams. Please check your teams on Moodle as, due to changes in course numbers, some groups have been reallocated between existing groups. If you have a new member please make them feel welcome.

We will design software using UML Class and Sequence diagrams, and perform a small amount of implementation work.

As a group you should design and implement **the following part** the Part Time Teachers Specification from AE1:

Before the start of each term or semester, the class directors produce a list of teaching requirements which we must try and fill. Our administrator will then attempt to find suitable staff and organise training for them.

You should use an Agile approach to designing your software and Trello to communicate between team members. You should design appropriate User Stories, but they do not have to be the same as your stories from AE1. Please only include stories related to the limited part of the software described above.

You should use sequence diagrams for each User Story to design your classes and methods, and produce a Class Structure Diagram. Each class should be owned by one team member who is responsible for implementing it (Note: members may be required to implement more than one class depending on your design). I suggest plantuml (<https://plantuml.com/>) as a quick way to draw UML diagrams, but you can use any approach you like, e.g. hand drawn, other tools etc.

Your stories will involve several different objects from different classes. Each class owner should implement the part of the story related to their class. This is how stories are split into tasks.

You should use a simple implementation and develop a standalone app that **does not require a real database**. All permanent information should be stored in a single file that is read in when the program starts and then written to when the program finishes. Remember your program will have an internal representation of this data, e.g. data-structures representing classes, teachers etc. Don't try and make the classes too elaborate as you don't have much time! These structures are empty when the program starts and then filled with data by reading from the file. The updated version of the data is then preserved by writing to the file before the program finishes. You should design this so that a better implementation, using a database, could be provided in release 2 with *minimum changes* to the code base. You **do not** need to write a database version for this exercise!

You will gain marks for using good design principles, e.g. ensuring low coupling and high cohesion, and using design patterns, covered in future weeks, when appropriate. Marks will also be awarded for working functionality.

Note: a well designed program where some things don't quite work can score just as many marks as a program that does everything but without good design principles.

### The Reports

Your **team report** should contain the following:

### **截止日期: 2023年3月17日星期五; 17:00**

这是评估练习1的后续行动，在可能的情况下，我们将使用相同的团队。请在Moodle上检查您的团队，因为由于课程编号的变化，一些组已在现有组之间重新分配。如果你有一个新成员，请让他们感到欢迎。

我们将使用UML类和序列图设计软件，并执行少量的实现工作。

作为一个小组，你应该设计和实现以下部分兼职教师规范从AE1:

在每个学期或学期开始之前，班主任会列出一份教学要求清单，我们必须尝试并填写。然后，我们的管理员将尝试寻找合适的员工并为他们组织培训。

您应该使用敏捷方法来设计您的软件和Trello，以便在团队成员之间进行通信。您应该设计适当的用户故事，但它们不必与AE1中的故事相同。请只包括与上述软件的有限部分有关的故事。

您应该为每个用户情景使用序列图来设计类和方法，并生成类结构图。每个类应该由一个负责实现它的团队成员拥有（注意：根据您的设计，成员可能需要实现多个类）。我建议plantuml (<https://plantuml.com/>) 作为绘制UML图的快速方法，但您可以使用任何您喜欢的方法，例如手绘，其他工具等。

您的故事将涉及来自不同类的几个不同对象。每个类所有者都应该实现与其类相关的故事部分。这就是故事被分成任务的方式。

您应该使用一个简单的实现，并开发一个不需要真正数据库的独立应用程序。所有永久信息应存储在一个文件中，该文件在程序启动时读入，然后在程序完成时写入。请记住，您的程序将具有此数据的内部表示，例如表示班级，教师等的数据结构。不要试图让课程过于复杂，因为你没有太多的时间！这些结构在程序启动时为空，然后通过从文件中读取来填充数据。然后，在程序完成之前，通过写入文件来保留数据的更新版本。您应该这样设计，以便在版本2中提供更好的实现，使用数据库，并对代码库进行最小的更改。您不需要为此练习编写数据库版本！

您将获得使用良好设计原则的分数，例如确保低耦合和高内聚，并在未来几周内适当时使用设计模式。工作功能也将获得分数。

注意：一个设计良好的程序，有些东西不太起作用，它的分数和一个做任何事情但没有好的设计原则的程序一样多。

### 报告

您的团队报告应包含以下内容:

- Your team name and a list of each team member's contribution including classes they were responsible for.
- A list of User Stories, as in AE1.
- Your class structure diagram.
- The sequence diagrams for each of the user stories.
- All the code from your program together with screenshots showing it running. We will run each of your programs to see how well they work. Please add comments to your code so we can understand what you were intending so that you can get marks if something doesn't quite work.
- A retrospective on what went well and what was challenging while completing this assignment.

As in AE1, please **also provide an individual report** on your team members and their contributions. **Each member should submit this separately.**

### **Report Naming Conventions (Important)**

For your **team report**, please use a filename of the form:

<GroupNum>\_SEIT\_AE2\_GroupReport.pdf

For example:

Group24\_SEIT\_AE2\_InduvidualReport.pdf

For your **individual report**, Please use a filename of the form:

<GUID>\_SEIT\_AE2\_IndividualReport.pdf

For example:

1002105a\_SEIT\_AE2\_IndividualReport.pdf

If you do not use this filename convention, we reserve the right to treat your reports as if they were unsubmitted.

•您的团队名称和每个团队成员的贡献列表，包括他们负责的班级。

\*用户情景列表，如AE1。\*你的类结构图。•每个用户故事的序列图。•程序中的所有代码以及显示其运行的屏幕截图。我们将运行您的每个程序，看看它们的工作情况。请在您的代码中添加注释，这样我们就可以理解您的意图，这样如果某些事情不太起作用，您就可以得到标记。

\*在完成这项作业时，回顾什么进展顺利，什么是挑战。

与AE1一样，请提供关于您的团队成员及其贡献的个人报告。每个成员应单独提交。

### **报告命名约定（重要）**

对于您的团队报告，请使用表格的文件名:

<GroupNum>\_SEIT\_AE2\_GroupReport.pdf

For example:

Group24\_SEIT\_AE2\_InduvidualReport.pdf

对于您的个人报告，请使用表格的文件名:

<GUID>\_SEIT\_AE2\_IndividualReport.pdf

For example:

1002105a\_SEIT\_AE2\_IndividualReport.pdf

如果您不使用此文件名约定，我们保留将您的报告视为未提交的权利。