Dr. Volker Riediger
M.Sc. Mahmood Al-Doori
M.Sc. Veronika Vasileva

university of koblenz
Computer Science

— Engineering Web and Data-intensive Systems - Winter Term 2022/23 —

Assignment 1.

From: Nov 07th, 2022 To: Nov 14th, 2022

Notes on the sumbission

Please store your solution in the OLAT group folder. We have already created a `solutions` folder for you.

Create a sub-folder for each assignment according to the scheme `solutions/assignmentNN`.

Put all solution files into that subfolder. We prefer PDF documents. You shall also add other files contributing to the solution, for example Astah models, database scripts (in later assignments).

Serious Advice

If you repeat the EWADIS course, we *strongly recommend* to redo the assignments from scratch. It is not very useful to copy solutions from the last semester, even if you think you have a perfect solution or a reference solution that we provided.

The intent of the assignments is to actively practice, not to present a perfect solution!

1 Architecture - Logical View - Domain Model

Part of the **logical view** of an architecture description is the so-called *domain model*.

Domain models deal with important concepts of an application, i.e., business objects and their relations. In a commercial project, the domain model represents the view of the customer on a system's data. It is the base for more detailed and more technical (implementation) data models.

Domain models are of Entity-Relationship (ER) type. There exist many ER languages, more or less expressive. A suitable diagram language for such models is a UML class diagram.

In principle, you could also use a sketching tool like Visio or draw.io to create the class diagram. Unlike in a simple drawing, in a UML model the various diagrams are *linked* to each other. Using a UML modeling tool enables consistent reuse of the model elements. Model elements can be used and viewed in more than one diagram. For example, if you rename a class, this change is automatically propagated to all diagrams that show the class.

Task:

Create a UML class diagram as a domain model for the OLAT system from a student's perspective.

Of course, this system is way too complex to be dealt with in a one-week assignment. Please restrict the domain model to the entities and relations that are currently used in the OLAT course for this class.

Remarks:

We expect a solution with around 10 classes with some attributes and their relations.

Please submit your solution as PDF file in your group folder under `solutions/assignment01`.

Additionally, please add the *source file* for your model.

We *recommend* to use the Astah modeling tool (see forum posts for additional information).

If you use Astah, please submit the model file `*.asta` in addition to the PDF.

If you are not familiar with UML class diagrams, we recommend the text book *UML @ Classroom* and the online resources related to that book (see link list in OLAT).

Please prepare to discuss your solution in one of the next tutorials!

The goals of this task are:

- Practice to use the Astah tool (we will use it throughout the whole course)
- Recap your UML knowledge
- Analyze an existing web application and reverse-engineer a domain model
- Review modeling conventions (see the UML reference card)
- Prepare the running example of this lecture
- We get an impression of your modeling skills

Non-Goals:

- You don't need to provide a *comprehensive* model
- You don't need to represent *all* features on detailed level
- You don't need to add *operations* to the classes, it shall only contain data items, their properties with appropriate types, and their relations