Model Driven Engineering AA 2023-2024

Document version: v2.0 13-Nov-2023

Prof. Alfonso Pierantonio
http://disim.univaq.it/AlfonsoPierantonio

Scope

The scope of the assignments for the Model-Driven Engineering course is to provide students with hands-on experience in using model-driven engineering techniques and tools to solve software engineering problems. This includes designing and implementing domain-specific modeling languages, creating models, developing model transformations, generating code or artifacts from models, and, more in general, building an ad-hoc modeling environment.

The assignments for the Model-Driven Engineering course involve several activities, including:

1. Analyzing and formalizing an applicative domain
2. Designing and implementing domain-specific modeling languages (DSMLs) using tools like Meta Programming System (MPS), Eclipse Modeling Framework (EMF)
3. Creating models using the defined DSMLs to represent different “systems” or domain instances
4. Transforming models between different representation formats or levels of abstraction
5. Applying model-driven engineering principles to generate code or other artifacts from models
6. Developing model transformations using languages like ATL (Atlas Transformation Language) or Epsilon.
7. Developing code generators using template-based languages like Acceleo and EGL
8. Analyzing and validating models for consistency, completeness, and correctness.
9. Collaborating with other students to work on relatively larger-scale modeling projects.

Applicative Domain

An application domain is the *segment of reality* for which we intend to formalize a metamodel. It is the background or starting point for the assignments described in this document.

The concept of an application domain is at least as comprehensive so that the domain concepts and relations relevant to the construction of models can be well understood

during the domain analysis. On the other hand, its extent should always be limited, that is, never be too complex.
An application domain typically includes a domain-specific language. This means that people in this domain use specific terms and concepts and think about their domain in a specific way.

A0. Assignment #0 (Applicative Domain)
You have to identify an applicative domain of your choice. It can be anything provided that you have some expertise about it.

Examples of domains are the following:

- E-Healthcare Process Management

  As in many knowledge-intensive domains, processes describing how the activities are scheduled among multiple actors are considered commonplace in healthcare. Being able to formalize such processes can provide the actors with exact and semi-formal descriptions for the correct process enactment, ie. the correct scheduling of tasks besides the behavior of each individual task and the data profile that tasks are expecting to be consumed and produced, respectively. The models represent, therefore, precise descriptions to be used for communication purposes among the stakeholders, instructions about how to execute the process, and formal artifacts to be automatically processed to generate documentation and configurations (eg for the workflow engine, etc).

- Educational Public Building

  A university edifice contains classrooms, laboratories, info points, and more. They can be composed to precisely describe what is needed for running a university department with multiple degree programs. […]

This assignment is mandatory. However, it is preliminary and does not give any score.

Assignment Score: 0/110.

Submission deadline: November 16th, 12:00 (new deadline)

A1. Assignment #1 (Class Diagram)
Define a metamodel in terms of a UML Class Diagram on the genMyModel platform (https://app.genmymodel.com) that captures the domain provided in A0.

You have to deliver the class diagram of the domain in A0 and an object diagram describing at least two instances for testing the metamodel correctness and expressiveness.

Assignment Score: 10/110.

Submission deadline: November 21st,12:00

A2. Assignment #2 (MPS)
Formalize the metamodel in A1 describing the domain in A0 using MPS. You have to provide the

- Language Structure
- A syntax of your choice and the corresponding projectional editor
- A validation rule that captures a specific condition that is not admitted
- Code generator for producing html documentation (in one page)

Assignment Score: 30/110

Submission deadline: December 5th,12:00

A3. Assignment #3 (EMF/1)
Formalize the metamodel given in A1 describing the domain in A0 by using EMF. You have to provide the

- The Ecore metamodel
- An OCL validation (as in A2)

Assignment Score: 20/110

Submission deadline: December 13th,12:00
A4. Assignment #4 (EMF/2)
Given the metamodel defined in A3, develop the following transformations:

- A transformation Model-to-Model in ATL or ETL
- A transformation Model-to-Code in Acceleo or EGL

Assignment Score: 20/110

Submission deadline: December 27th,12:00

A5. Assignment #5 (EMF/Concrete Syntax)
Given the metamodel in A3, define a visual notation and a textual notation for it with the following frameworks

- Sirius
- Xtext

Assignment Score: 25/110

Submission deadline: January 10th,12:00

Assignment Submission Procedure

Assignment A0

Please use the following link to enter

- the information about your team
- the domain description

by using the following link:
https://forms.office.com/Pages/DesignPageV2.aspx?origin=NeoPortalPage&subpage=design&id=fIrwndcxJECbpl7V76waATtCZR6UOzNFstta9rEGzxlUMjdJU1NMQUVUV1hLVTZBS0lGVVlKSU5NWi4u

Important: even if you have already uploaded your domain description, please enter it once again so that we can overwrite the previous one, as most of you have refined/modified the original one.

Assignment A1-A5

All the assignments must be submitted by the corresponding deadline through GitHub Classroom. GitHub Classroom is a powerful tool for managing coding assignments in an educational setting, leveraging the collaborative and version control features of Git and GitHub.

What you have to do?

1. go to https://classroom.github.com/a/TVqxPKtc
2. create your team
3. accept the next assignment

at this point, you will be given a repository that will accept commits until the deadline expires.