

Model-Driven Software Engineering

MO631 – Engenharia de Software Dirigida por Modelos

MC976(B) – Tópicos em Engenharia de Software II

2º Semestre 2019

Prof. Leonardo Montecchi
<http://ic.unicamp.br/~leonardo>

| | Dia | Horário |
|--------------------|--------------------------------------------------------------------------|----------------|
| Aula | Segunda-feira | 16h-18h |
| | Quarta-feira | 16h-18h |
| Atendimento | Alunos podem marcar horário para atendimento com o professor por e-mail. | |

Objetivo

Ao final da disciplina o aluno deverá ser capaz de aplicar as principais etapas da engenharia dirigida por modelos: construir metamodelos, projetar e implementar transformações de modelos, e construir um gerador de código.

Conhecimentos Requeridos

- Princípios da programação orientada a objetos: classe, instância, herança, associação, composição, etc.
- Ambiente de programação Java

Conteúdo

A disciplina é organizada em seis unidades:

1. *Introdução e Metodologia*: Modelos, metamodelos, hierarquia de meta-modelagem. Engenharia dirigida por modelos. Taxonomia, principais aplicações, e metodologia geral.
2. *Linguagens Específicas de Domínio*: Linguagens específicas de domínio e suas principais características. Construir um metamodelo. Conceito de espaço técnico. MDA e os padrões da OMG. Meta-modelagem vs. perfis UML. Criação de um perfil UML personalizado.
3. *Eclipse Modeling Framework*: Introdução à plataforma Eclipse para modelagem. A linguagem Ecore. Serialização. Geração básica de código. Dualidade modelo-código.
4. *Especificação de Vínculos e Buscas*: OCL. Utilizo de OCL em Ecore. Tradeoff entre vínculos e metamodelos mais rigorosos.
5. *Transformação de Modelos*: Taxonomia das transformações de modelos. Padrão QVT. ATL. Estrutura de uma regra de transformação. Rastreabilidade. Tipos de regras: rules, called rules, lazy rules, unique lazy rules. Medição de modelos.
6. *Geração de Código*: Geração de código como caso particular de transformação. Cadeias de transformações. Geração baseada em templates. Aceleo. Geração de arquivos múltiplos. Interação com código Java.

Avaliação

A avaliação da disciplina será realizada com base em um teste teórico (T) e um projeto (J). O teste teórico é individual e vale 20% da nota final; o projeto pode ser desenvolvido em duplas e vale 80% da nota final.

O projeto consiste em três atividades:

- J_1 : Definição de metamodelo(s) – (40% da nota do projeto)
- J_2 e J_3 : Duas atividades escolhidas entre i) transformação text-to-model, ii) transformação model-to-model, e iii) transformação model-to-text – (30% da nota do projeto cada)

A média M da disciplina será calculada então como:

$$M = 0.2T + 0.8(0.4J_1 + 0.3J_2 + 0.3J_3)$$

Serão sugeridos alguns temas para o projeto, mas outros temas poderão ser combinados com o professor nas primeiras semanas da disciplina, sendo com preferência um tema relacionado ao tópico de pesquisa (ou de interesse) do aluno.

Datas

Estão marcadas duas entregas parciais do projeto (obrigatórias) para monitorar o progresso. Contudo, o projeto pode ser refinado ao longo da disciplina e **as notas serão definidas com base na entrega final.**

Compativelmente com o número de alunos na turma, os projetos serão apresentados no Workshop dos Trabalhos de Pós do IC, que acontecerá dia 27 de novembro, junto com o Workshop de Teses e Dissertações (WTD).

As datas definidas a seguir podem sofrer alterações devido a imprevistos ou necessidades relacionadas ao desenvolvimento da disciplina.

| | |
|-------------------|------------------------------------------------------------------|
| 27/09/2019 | Entrega parcial projeto |
| 25/10/2019 | Entrega parcial projeto |
| 30/10/2019 | Teste Teórico |
| 27/11/2019 | Apresentação dos projetos no Workshop dos Trabalhos de Pós do IC |
| 29/11/2019 | Entrega final projeto |

Observações

- **As aulas e as outras atividades da disciplina serão em inglês.** Esclarecimento de dúvidas e observações poderão ocorrer em português.
- A presença é obrigatória em todas as aulas. Frequência inferior a 75% causa reprovação.

Referências

Não há um livro texto específico. Os seguintes livros cobrem o que será visto em aula. Demais material sobre assuntos específicos será fornecido ao longo da disciplina.

- T. Stahl, M. Völter, Model-Driven Software Development, John Wiley & Sons, 2006.
- M. Brambilla, J. Cabot, M. Wimmer, Model-Driven Software Engineering in Practice, Second Edition, Morgan & Claypool Publishers, 2017.
- D. Steinberg, F. Budinsky, M. Paternostro, E. Merks, EMF: Eclipse Modeling Framework, Second Edition, Addison-Wesley, 2009.