## **SYLLABUS**

# Behavior Modeling of Software Systems

# University year 2025

# 1. Information regarding the programme

1.1. Higher education institution	Babeş Bolyai University
1.2. Faculty	Faculty of Mathematics and Computer Science
1.3. Department	Department of Computer Science
1.4. Field of study	Computer Science
1.5. Study cycle	Master
1.6. Study programme/Qualification	Software Engineering
1.7. Form of education	

#### 2. Information regarding the discipline

2.1. Name of the dis	scipli	ne <b>Behavior</b>	Behavior Modeling of Software Systems			Discipline code	MME8006		
2.2. Course coordinator					Lect. dr. Ioan Lazar				
2.3. Seminar coordinator				Lect. dr. Ioan Lazar					
2.4. Year of study	1	2.5. Semester	2	2.6. Type of evaluation	on	С	2.7. Dis	cipline regime	Optional

#### 3. Total estimated time (hours/semester of didactic activities)

3.1. Hours per week	3	of which: 3.2 course	2	3.3 seminar/laboratory/project	1
3.4. Total hours in the curriculum	42	of which: 3.5 course	28	3.6 seminar/laboratory/project	14
Time allotment for individual study (ID) and self-study activities (SA)					
Learning using manual, course support,	bibliogr	aphy, course notes (SA)			28
Additional documentation (in libraries, on electronic platforms, field documentation)					14
Preparation for seminars/labs, homework, papers, portfolios and essays					14
Tutorship					
Evaluations					
Other activities:					14
3.7. Total individual study hours 22					
3.8. Total hours per semester	120				
3.9. Number of ECTS credits	mber of ECTS credits 7				

#### 4. Prerequisites (if necessary)

4.1. curriculum	Programming Fundamentals
4.2. competencies	• Good programming skills in at least one of the languages Java, C#

## 5. Conditions (if necessary)

5.1. for the course	
5.2. for the seminar /lab activities	

6.1. Spe	cific competencies acquired <sup>1</sup>
Pro fess ion al/ ess enti al co mp ete nci	<ul> <li>C 4.3 Identify models and methods adequate to real life problem solving</li> <li>C 2.1 Identify adequate software systems development methodologies</li> <li>C 1.1 Proper description of programming paradigms and language specific mechanisms, and identification of semantical an syntactical differences</li> </ul>
Tra nsv ers al co mp ete nci es	<ul> <li>CT1 Apply organized and efficient work rules and responsible attitude towards didactical and research field, in order to creatively use work potential; respect professional ethical principles</li> <li>CT3 Use efficient methods and techniques for: learning, information search, research and development of capacities to adapt to the requirements of a dynamic society and to communicate in an international language</li> </ul>

## 6.2. Learning outcomes

Kno wle dge	The student knows: development cycle of systems based on services
Skil ls	The student is able to develop a system based on services
Respo nsibili ty and auton omy:	The student has the ability to work independently to build SOA systems

7. Objectives of the discipline (outcome of the acquired competencies)

<sup>&</sup>lt;sup>1</sup> One can choose either competences or learning outcomes, or both. If only one option is chosen, the row related to the other option will be deleted, and the kept one will be numbered 6.

	<ul> <li>Enhance the students understanding of behavior modeling concepts through a practical and pragmatic approach</li> </ul>
7.1 General objective of the discipline	<ul> <li>Provide the students with an environment in which they can explore the usage and usefulness of behavior modeling concepts in various business scenarios</li> </ul>
	<ul> <li>Induce a realistic and industry driven view of software design concepts such as design patterns and their inherent benefits</li> </ul>
7.2 Specific objective of the discipline	<ul> <li>Give students the ability to explore various object oriented programming languages</li> <li>Improve the students abilities to tackle business requirements</li> <li>Enhance the students understanding of state machines, activities</li> <li>Provide students with insights into the way of working towards achieving high quality software through skilled trainers from the IT industry</li> </ul>

#### 8. Content

8.1 Course	Teaching methods	Remarks
<ol> <li>Introduction to DSML</li> <li>Textual notations</li> <li>Introduction to DSML</li> <li>Graphical notations</li> <li>Model Transformations in the Context of MDA/UML</li> <li>MDA, UML</li> <li>M2T, M2M</li> </ol>		
<ul> <li>4. Service Component Models</li> <li>Pltaform specific frameworks: iPOJO, SCA</li> <li>UML: deployment diagrams, composite</li> <li>structure diagrams</li> <li>5. Service oriented architecture Modelling</li> <li>Language (SoaML)</li> <li>SoaML specification</li> <li>ModelPro/MagicDraw frameworks</li> <li>6. Business Processes</li> <li>Business Process Modelling Notation (BPMN)</li> <li>Workflow Patterns</li> </ul>	Exposure: description, explanation, examples, discussion of case studies	
7. Business Processes Workflow Patterns 8. Foundational UML Abstract Syntax and Foundational Model Library Java to UML activity mapping		
9. Action Language for Foundational UML (Alf)		

10. Modeling user interface navigation using state machines UML: state machine diagrams					
<ul> <li>11. Modeling user interface navigation using state machines</li> <li>Platform specific frameworks: Grails and JBoss Seam</li> <li>12. Capturing requirements with Business</li> <li>Motivation Model</li> <li>BMM</li> <li>13. Capturing requirements with Business</li> <li>Motivation Model</li> <li>From BMM to SOA</li> </ul>					
14. Executable Use Cases					
Bibliography [Ambler04] Ambler, S.W. The Object Primer: Agil Press, 2004. [Fowler99] Fowler, M. Analysis Patterns - Reusab [Evans03] Evans, E. Domain-Driven Design: Tack [OMG03] OMG. MDA Guide Version 1.0.1. Object 1 http://www.omg.org/docs/omg/03-06-01.pdf [OMG06] OMG. Business Process Modeling Notat http://www.bpmn.org/ [WPI06] Workflow Patterns Initiative. Control-Fl Patterns Home Page. 2006. http://www.workflow	Bibliography [Ambler04] Ambler, S.W. The Object Primer: Agile Model-Driven Development with UML 2.0. Cambridge University Press, 2004. [Fowler99] Fowler, M. Analysis Patterns - Reusable Object Models. Addison-Wesley, 1997. [Evans03] Evans, E. Domain-Driven Design: Tackling Complexity in the Heart of Software. Addison-Wesley, 2003. [OMG03] OMG. MDA Guide Version 1.0.1. Object Management Group, 2003. http://www.omg.org/docs/omg/03-06-01.pdf [OMG06] OMG. Business Process Modeling Notation Specification, V1.0. Object Management Group, 2006. http://www.bpmn.org/ [WPI06] Workflow Patterns Initiative Control-Flow Data Resource and Exception Handling Patterns Workflow				
8.2 Seminar / laboratory	Teaching methods	Remarks			
<ol> <li>Introduction to DSML</li> <li>Model Transformations in the Context of MDA/UML</li> <li>Service Component Models</li> <li>Service oriented architecture Modelling Language (SoaML)</li> <li>Business Processes</li> <li>Workflow Patterns</li> <li>Foundational UML</li> </ol>	Dialogue, debate, case studies, examples, proofs				
7. Foundational UML					

# 9. Corroborating the content of the discipline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

- The course respects the IEEE and ACM Curriculla Recommendations for Computer Science studies;
- The course exists in the studying program of all major universities in Romania and abroad;
- The content of the course is considered by the software companies as important for average programming skills.

## 10. Evaluation

Activity type	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Percentage of final grade		
10.4 Course	<ul> <li>know the basic principle of the domain;</li> <li>apply the course concepts</li> <li>problem solving</li> </ul>	Written exam	50%		
10.5 Seminar/lab activities	<ul> <li>be able to implement course concepts and algorithms</li> <li>apply techniques for different classes of programming languages</li> </ul>	-Practical examination -documentation -portofolio -continous observations	50%		
10.6 Minimum standard of performance					
<ul> <li>A minimum passing grade is defined by attaining at least 50% (5/10) points for the final project and each of the three lab assignments respectively.</li> <li>No more than 3 absences are allowed for the seminar/lab activities</li> </ul>					

# 11. Labels ODD (Sustainable Development Goals)<sup>2</sup>

Not applicable.

<sup>&</sup>lt;sup>2</sup> Keep only the labels that, according to the *Procedure for applying ODD labels in the academic process*, suit the discipline and delete the others, including the general one for *Sustainable Development* – if not applicable. If no label describes the discipline, delete them all and write *"Not applicable."*.

Date: .30.04.2025 Signature of course coordinator

Lect. dr. Ioan Lazar

Signature of seminar coordinator

Lect. dr. Ioan Lazar

Date of approval:

...

Signature of the head of department Assoc.prof.phd. Adrian STERCA