# SYLLABUS

*Agile Project Management*

University year 2025-2026

**1. Information regarding the programme**

| | |
|---|---|
| 1.1. Higher education institution | Babeş-Bolyai University, Cluj Napoca |
| 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 | Cyber Security |
| 1.7. Form of education | Optional |

**2. Information regarding the discipline**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2.1. Name of the discipline | **Agile Project Management** | | | | | Discipline code | **MME8193** |
| 2.2. Course coordinator | | | | Assoc. prof. phd. Dan Mircea SUCIU | | | |
| 2.3. Seminar coordinator | | | | Assoc. prof. phd. Dan Mircea SUCIU | | | |
| 2.4. Year of study | 1 | 2.5. Semester | 1 | 2.6. Type of evaluation | E | 2.7. Discipline regime | Compulsory |

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

| | | | | | |
|---|---|---|---|---|---|
| 3.1. Hours per week | **4** | of which: 3.2 course | **2** | 3.3 seminar/laboratory/project | **2** |
| 3.4. Total hours in the curriculum | **56** | of which: 3.5 course | **28** | 3.6 seminar/laboratory/project | **28** |

| **Time allotment for individual study (ID) and self-study activities (SA)** | **hours** |
|---|---|
| Learning using manual, course support, bibliography, course notes (SA) | 15 |
| Additional documentation (in libraries, on electronic platforms, field documentation) | 20 |
| Preparation for seminars/labs, homework, papers, portfolios and essays | 20 |
| Tutorship | 4 |
| Evaluations | 10 |
| Other activities: | |

| | |
|---|---|
| **3.7. Total individual study hours** | **69** |
| **3.8. Total hours per semester** | **125** |
| **3.9. Number of ECTS credits** | **5** |

**4. Prerequisites** (if necessary)

| | |
|---|---|
| 4.1. curriculum | - |
| 4.2. competencies | - |

**5. Conditions** (if necessary)

| | |
|---|---|
| 5.1. for the course | Video projector |
| 5.2. for the seminar /lab activities | Video projector |

**6.1. Specific competencies acquired** [1]

[1] 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.

| | |
|---|---|
| **Professional/essential competencies** | • advanced knowledge of theoretical, methodological, and practical developments in computer science;<br>• proficient use of verification, validation, and evaluation criteria and methods to his/her own software solutions, ability to formulate value judgements and to justify/explain constructive decisions; |
| **Transversal competencies** | • systematic use of computer science knowledge to model and interpret new situations, within application contexts larger than the known ones;<br>• detailed knowledge and integrated use of conceptual and methodological apparatus pertaining to informatics to provide solutions for incompletely defined situations, to solve new theoretical and practical problems; |

**6.2. Learning outcomes**

| | |
|---|---|
| **Knowledge** | • The graduate possesses the fundamental knowledge for modelling, being able to analyse real life problems and to translate them in concrete requirements and to design a corresponding software model |
| **Skills** | • The graduate proofs working skills in professional teams an interdisciplinary in order to efficiently implement programmes and research programmes in computer science<br>• The graduate can use specific language and terminology for databases domain being able to communicate and interact with members of a team |
| **Responsibility and autonomy:** | • The graduate uses efficient strategies, methods and techniques for lifelong education, in order to self educate and self develop his/her personal and professional skills<br>• The graduate has the ability to combine information in different ways in order to form a positive attitude towards its his/her own development |

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

| | |
|---|---|
| **7.1 General objective of the discipline** | • acquiring the knowledge and skills required for an effective IT project management process |
| **7.2 Specific objective of the discipline** | • analysis of the factors that contribute to the superiority of Agile methodologies over predictive approaches in software project management<br>• evaluation of the strengths and limitations of current Agile practices<br>• identification and understanding of the software project lifecycle within an Agile framework |

**8. Content**

| 8.1 Course | Teaching methods | Remarks |
|---|---|---|
| 1. Introduction to Agile Methods for Software | Interactive exposure, | |

| | | |
|---|---|---|
| Project Development | Explanation, Conversation, Didactical demonstration | |
| 2., 3., 4. Scrum Methodology – Roles, Artifacts, Ceremonies | Interactive exposure, Explanation, Conversation, Didactical demonstration | |
| 5., 6. Extreme Programming Methodology – Values, Principles, Practices | Interactive exposure, Explanation, Conversation, Didactical demonstration | |
| 7. Lean Software Development Approach | Interactive exposure, Explanation, Conversation, Didactical demonstration | |
| 8., 9. Kanban Methodology | Interactive exposure, Explanation, Conversation, Didactical demonstration | |
| 10., 11. Other Agile Methodologies: DSDM, Crystal, Feature-Driven Development, Agile Unified Process, Disciplined Agile, SAFe | Interactive exposure, Explanation, Conversation, Didactical demonstration | |
| 12. Estimation in Agile Projects | Interactive exposure, Explanation, Conversation, Didactical demonstration | |
| 13. Agile Contracts. Risk Management in Agile Projects | Interactive exposure, Explanation, Conversation, Didactical demonstration | |
| 14. Performance Management in Agile Projects | Interactive exposure, Explanation, Conversation, Didactical demonstration | |
| Bibliography<br>1. Jeff Langr, Tim Ottinger - Agile in a Flash: Speed-Learning Agile Software Development, Pragmatic Bookshelf, 2011<br>2. Esther Derby, Diana Larsen - Agile Retrospectives: Making Good Teams Great, Pragmatic Bookshelf, 2006<br>3. Thomas Stober, Uve Hansmann - Agile Software Development, Best Prectices for Large Software Development Projects, Springer 2010<br>4. Mike Cohn - Succeeding with Agile Software Development using Scrum, Addison-Wesley, 2010<br>5. Mike Cohn - User Stories Applied, For Agile Software Development, Addison-Wesley, 2004 | | |
| 8.2 Seminar / laboratory | Teaching methods | Remarks |
| Agile problem solving | Guided discussions, simulations, educational games | |
| Implementing Agile principles | Guided discussions, simulations, educational games | |
| Agile estimation techniques | Guided discussions, simulations, educational games | |
| Task prioritization | Guided discussions, simulations, educational games | |
| Self-organizing teams | Guided discussions, simulations, educational games | |
| Delegation in Agile teams | Guided discussions, simulations, educational games | |
| Agile coaching | Guided discussions, simulations, educational games | |
| Bibliography<br>1. Tom Demarco - Waltzing with Bears Managing Risks On Software Projects<br>2. Patrick Lencioni - The Five Dysfunctions of a Team, Jossey-Bass, 2002<br>3. Daniel Goleman - Leadership: The Power of Emotional Intellegence, More Than Sound, 2011 | | |

**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 content of this course is aligned with the practices and recommendations of international professional communities such as Agile Alliance, Scrum.org, and PMI-ACP. It addresses the current needs and expectations of software industry employers, particularly in areas such as incremental delivery, team collaboration, continuous improvement, and adaptability in software projects.

The course covers concepts and methods that are widely used in real-world Agile environments, including frameworks like Scrum, Kanban, and Extreme Programming, which are commonly adopted by both local and international IT companies.

Additionally, the course reflects current trends in academic research related to modern software engineering and product development in dynamic, uncertain environments, where agility, collaboration, and continuous delivery are critical success factors.

**10. Evaluation**

| Activity type | 10.1 Evaluation criteria | 10.2 Evaluation methods | 10.3 Percentage of final grade |
|---|---|---|---|
| 10.4 Course | - know the basic principles of the domain;<br>- apply the course concepts<br>- problem solving | Completion of individual missions that will be activated weekly | 80% |
| 10.5 Seminar/laboratory | - problem solving in Agile teams | - oral examination<br>- continuous observations | 20% |
| 10.6 Minimum standard of performance | | | |
| • The final grade must be a minimum of 5 in order to pass. | | | |

**11. Labels ODD (Sustainable Development Goals)[2]**

*Not applicable*.

| Date: | Signature of course coordinator | Signature of seminar coordinator |
|---|---|---|
| 15.04.2025 | Assoc. prof. phd. Dan Mircea SUCIU | Assoc. prof. phd. Dan Mircea SUCIU |

Date of approval:
...

Signature of the head of department

Assoc. prof. phd. Adrian STERCA

[2] 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*.”.