# COURSE DESCRIPTION

*Embedded Systems: Applications and Cybersecurity - Computational Models for Embedded Systems*

Academic year 2026-2027

**1. Programme-related data**

| | |
|---|---|
| 1.1. Higher Education Institution | Babes-Bolyai University |
| 1.2. Faculty | Faculty of Mathematics and Computer Science |
| 1.3. Department | Computer Science |
| 1.4. Field | Computer Science |
| 1.5. Level of study | Master |
| 1.6. Degree programme / Qualification | Software engineering |
| 1.7. Form of education | Full time |

**2. Course-related data**

| | | | | | |
|---|---|---|---|---|---|
| 2.1. Course title | **Embedded Systems: Applications and Cybersecurity - Computational Models for Embedded Systems** | | | Course code | **MME8245** |
| 2.2. Course coordinator | Assoc. Prof. dr. Vescan Andreea | | | | |
| 2.3. Seminar coordinator | Assoc. Prof. dr. Vescan Andreea | | | | |
| 2.4. Year of study | 2 | 2.5. Semester | 3 | 2.6. Type of assessment | Exam |
| 2.7. Course status | Optional | | | 2.8. Course type | Specialisation subject |

**3. Total estimated time** (hours per semester of teaching activities)

| | | | | | |
|---|---|---|---|---|---|
| 3.1. Number of hours per week | **4** | of which: 3.2. course | **2** | 3.3. seminar/ laboratory/ project | **2** |
| 3.4. Total of hours in the curriculum | 56 | of which: 3.5. course | 28 | 3.6. seminar/ laboratory | **28** |
| **Time allocation for individual study (IS) and self-taught activities (ST)** | | | | | **hours** |
| Learning from textbooks, course materials, bibliography, and notes (IS) | | | | | 84 |
| Additional research in the library, on subject-specific electronic platforms, and on-site | | | | | 14 |
| Preparing seminars/ laboratories/ projects, assignments, reports, portfolios, and essays | | | | | 14 |
| Tutoring (professional guidance) | | | | | 3 |
| Examinations | | | | | 4 |
| Other activities | | | | | 0 |
| **3.7. Total hours of individual study (IS) and self-taught activities (ST)** | | | | **119** | |
| **3.8. Total hours per semester** | | | | **175** | |
| **3.9. Number of credits** | | | | **7** | |

**4. Prerequisites** (where applicable)

| | |
|---|---|
| 4.1. curriculum-related | |
| 4.2 skills-related | |

**5. Specific conditions** (where applicable)

| | |
|---|---|
| 5.1. course-related | Video projector, Internet access |
| 5.2. seminar/laboratory-related | Laboratory with computers; model checking tools, FSM/PN tools |

**6.1. Competencies resulting from the completion of the degree programme *(as referred to in the curriculum)*[1]**

| Professional competencies | |
|---|---|
| **Competency code** | **Competency** |
| **PC1** | understanding and working with basic concepts in software engineering |
| **PC3** | assimilation of mathematical concepts and formal models to understand, verify and validate software systems; |
| **PC5** | proficient use of methodologies and tools specific to software engineering |
| **Transversal competencies** | |
| **Competency code** | **Competency** |
| **TC1** | capability of information analysis and synthesis; etic and fair behaviour, commitment to professional deontology |
| **TC2** | team work capabilities, and ability to fulfil different roles |
| **TC3** | professional communication skills; concise and precise description, both oral and written, of professional results |

**6.2. Learning outcomes relevant to the degree programme** (as referred to in the curriculum)[2]

| Learning outcomes targeted by the subject | | |
|---|---|---|
| **Competency code** | **Knowledge and comprehension** | **Specific academic skills** |
| **PC1** | The graduate has the necessary knowledge to devise, model and design of complex software applications. | The graduate is able to carry on activities for education and training on different topics related to software systems. |
| **PC3** | The graduate has the skills to perform research in software engineering especially for algorithmic thinking and for critical thinking. | The graduate can use specific language and terminology for the field of software engineering being able to communicate and interact with members of a team. |
| **PC5** | The graduate can apply advanced software engineering knowledge starting from a high level of abstraction and being able to offer implementation solutions for complex software systems. | The graduate knows and respects the ethical and legal principles and rules in scientific research. |
| **CT1** | 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. | The graduate proves knowledge related to specifying the requirements of research activities in the domain of computer science in general and software engineering in particular and he/she understands the role of research in promoting progress. |
| **CT2, CT3** | The graduate has the skills to perform research in software engineering especially for algorithmic thinking and for critical thinking. | The graduate can use specific language and terminology for the field of software engineering being able to communicate and interact with members of a team. |

**7. Subject-specific learning outcomes**

| Knowledge and comprehension |
|---|

[1] The professional and/or transversal skills targeted by the subject for which the course description is prepared will be copied from the curriculum of the degree programme. For each competency, the complete entry, including the competency code, will be copied with the exact wording that appears in the curriculum, without any changes. If no competency is copied from either of the two categories, the row corresponding to that category is deleted from the table.

[2] The learning outcomes relevant for the degree programme and targeted by the subject for which the course description is prepared will be listed. The entries, copied without any changes from the Curriculum by subject type (Core Subject/Specialisation Subject/Complementary Subject), are listed under the corresponding competency.

| |
|---|
| 1. The graduate is familiar with the concepts related to software modelling and is able to implement functional and non-functional requirements described in specific documents for the analysis and design of software systems. |
| 2. The graduate has the necessary skills to use research support tools. |
| **Specific academic skills** |
| 1. The graduate has the knowledge to apply model-based software development techniques. |
| 2. The graduate is able to present and explain methods, algorithms, paradigms and techniques used in various branches of computer science. |
| 3. The graduate is able to write a scientific/technical report. |

**8. Contents**

| **8.1. Course** | **Teaching and learning methods** | **Remarks[3]** |
|---|---|---|
| (1) Introduction. Model: Why? What? How? Types of systems. Requirements and Safety Requirements. | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| (2) Model checking | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| Model checking | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| (3) Synchronous models Asynchronous models. | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| Automotive Invited Lecture | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| IoT + Real time: | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| Finite State Machines (1)+(2) | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| Finite State Machines (1)+(2) | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| Petri nets | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| Timed models | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| Hybrid systems | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |

[3] For example, organisational aspects, recommendations for students, specific aspects relating to the course/seminar, such as inviting experts in the field, etc.

| | | |
|---|---|---|
| Dynamical systems | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| Research report presentation by students | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |
| Research report presentation by students | Interactive exposure<br>Explanation, Conversation<br>Didactical demonstration | |

Bibliography

Bibliography

**Books**

[Kat08] C. Baier, J.-P. Katoen, Principles of Model Checking, ISBN 978-0-262-02649-9, 2008

[Ari08] M. Ben-Ari, Principles of the Spin Model Checker, ISBN 978-1-84628-769-5, 2008

[Noe05] T. Noergaard, Embedded systems architecture: a comprehensive guide to engineers and programmers, Elsevier, 2005

[Hoa04] Hoare, CAR (2004) (1985), Communicating Sequential Processes, Prentice Hall International

[Pon02] M. Pont, Embedded C, Addison-Wesley, 2002

[Boo67] Taylor Booth (1967) Sequential Machines and Automata Theory, John Wiley and Sons, New York. Library of Congress Catalog Card Number: 67-25924.

**Articles**

[Har87] D. Harel, "Statecharts: A Visual Formalism for Complex Systems", Sci. Comput. Programming 8 (1987), 231-274

[Pet66] Petri, CA (1966) Communication with automata. DTIC Research Report AD0630125

**Tutorials**

During lectures/seminars/laboratories tutorials will be given for each assignment.

| **8.2. Seminar/ laboratory** | **Teaching and learning methods** | **Remarks** |
|---|---|---|
| Seminar 1, 2, 3<br>Model Cheking<br>Specifying safety and liveness requirements. | Presentation, Conversation, Problematizations, Discovery, Individual study, Exercises | |
| Seminar 4, 5, 6<br>Finite State Machines Project Activity<br>Using Finite State Machines or/and PetriNets to model an embedded system | Presentation, Conversation, Problematizations, Discovery, Individual study, Exercises | |
| Seminar 7<br>Delivery of projects (not delivered in Seminar 3 or Seminar 6) | Presentation, Conversation, Problematizations, Discovery, Individual study, Exercises | |

Remark:

- Students will search and use model cheking tools suitable for their Model Checking Project Activity. http://spinroot.com/spin/whatispin.html
- Students will search and use FSM/PN tools suitable for their FSM/PN Project Activity.

Bibliography

See from Courses content.

- This course follows 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;
  http://www.seas.upenn.edu/~cis540/
  https://inst.eecs.berkeley.edu/~ee249/fa07/
  http://www.ict.kth.se/courses/IL2202/
  http://users.abo.fi/lmorel/MoCs/
  http://bears.ece.ucsb.edu/class/ece253/

Course content is considered very important by the software companies for improving advance embedded systems modeling and verifying skills.

**9. Evaluation**

| Type of activity | 9.1 Evaluation criteria[4] | 9.2 Evaluation methods[5] | 9.3 Percentage in the final grade |
|---|---|---|---|
| 9.4. Course | The correctness and completeness of the accumulated knowledge of computational models for embedded systems. | Written exam (in the regular session)<br>InClass - Quiz<br>TakeHome - Evaluation of the research report documentation+presentation | 50% |
| 9.5. Seminar/ laboratory | Problem definition | Evaluation of investigation of the Problem definition | 10% |
| | Problem definition and specification in JSpin, Show that it is possible to reach the desired end state | Evaluation of the project (modeling, verification properties) | 20% |
| | Use Finite State Machine to model the embedded system. | Evaluation of the project (modeling, I/O, computational model used) | 20% |
| 9.6 Minimum standard for passing | | | |
| Each student has to prove that:<br>➢ (s)he acquired an acceptable level of knowledge and understanding of the computational models for embedded systems;<br>➢ (s)he has the ability to establish certain connections and to use the knowledge in solving different problems.<br>Successful passing of the exam is conditioned by the final grade that has to be at least 5. | | | |

**10. SDG labels (Sustainable Development Goals)[6]**

[4] The evaluation criteria must directly reflect the learning outcomes targeted at the level of the degree programme respectively at the level of the subject. More specifically, the learning outcomes set out in the expected learning outcomes are assessed.

[5] Both final evaluation methods and ongoing evaluation strategies should be established.

[6] Select a single label which, according to the *Implementation of SDG labels in the academic process*, best matches the subject. If the subject addresses sustainable development in a generic manner (i.e. by presenting/introducing

| | | |
|---|---|---|
| | ○ | Sustainable Development Generic Label |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
|  |  |  |  |  |  |  |  |  |
| ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ● |
|  |  |  |  |  |  |  |  | No label applies |
| ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |

Date of entry:
22.05.2026

Signature of course coordinator

Assoc. Prof. Vescan Andreea

Signature of seminar coordinator

Assoc. Prof. Vescan Andreea

Date of approval in the department:
...

Signature of the head of department

Assoc.prof.phd. Adrian STERCA

the general framework of sustainable development, etc.), then the Sustainable Development generic label may be applied. If none of the labels describe the subject, select the last option: *"No label applies."*