SYLLABUS

Virtual instrumentation

University year 2025-2026

1. Information regarding the programme

1.1. Higher education institution	Babeș-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	Bachelor
1.6. Study programme/Qualification	Computer Science
1.7. Form of education	Full time

2. Information regarding the discipline

2.1. Name of the discipline	Virtual instrumentation					Discipline code	**MLE5092**
2.2. Course coordinator	**Prof.Dipl.Eng.PhD. Horia Hedeșiu**						
2.3. Seminar coordinator	**Prof.Dipl.Eng.PhD. Horia Hedeșiu**						
2.4. Year of study	3	2.5. Semester	6	2.6. Type of evaluation	C	2.7. Discipline regime	Elective

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

3.1. Hours per week	**5**	of which: 3.2 course	**2**	3.3 seminar/laboratory/project	**3**
3.4. Total hours in the curriculum	60	of which: 3.5 course	24	3.6 seminar/laboratory/project	**36**

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

3.7. Total individual study hours	65
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	Class room with a video projector device, Internet connection
5.2. for the seminar /lab activities	• Computers • embedded systems myRIO • accessories for myRIO systems: sensors, displays, connectors, passive and active electronic components

6.1. Specific competencies acquired [1]

Professional/essential competencies	• development and maintenance of software systems • use of software tools in an interdisciplinary context
Transversal competencies	• Applying rules for an organized and efficient work, responsible attitude towards the didactic-scientific field for creative capitalization of one's own potential, complying to the principles and professional ethics norms. • Utilizing efficient methods and techniques for learning, knowing, research and development of knowledge capitalization capacities, adapting to the requirements of a dynamic society and the communication in Romanian or an international language.

6.2. Learning outcomes

Knowledge	• The graduate has the necessary knowledge for using computers, developing software programs and applications, information processing. • The graduate knows multiple programming languages and is able to write applications in compiled, interpreted or dynamic languages with the ability to choose the appropriate programming language for the specific application to be developed.
Skills	• The graduate has the ability to develop, design and create new applications, systems or products using best practices of the field. • The graduate has the ability to understand and use design patterns for application development.
Responsibility and autonomy:	• The graduate has the ability to observe and obtain information from various sources. • The graduate has the necessary knowledge to process and verify data and information.

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

7.1 General objective of the discipline	• Develop practical abilities to implement embedded systems, produce functional prototypes which may be used in applied research
7.2 Specific objective of the discipline	• Learning and understanding of the concepts and notions related to the graphical programming language G, respectively programming framework LabVIEW

[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.

8. Content

8.1 Course	Teaching methods	Remarks
1. Introduction to Virtual Instrumentation. System graphical design. Industrial Internet of Things	Exposure:description, explanation,examples, discussion of case studies	
2. Fundamentals of graphical programming in G 1/3 virtual instruments, VI – Front Panel, Block Diagram, Tool Palette, structures, clusters, debuggi error management	Exposure:description, explanation,examples, discussion of case studies	
3.Fundamentals of graphical programming in G 2/3: implement VI, document graphical code, timing issues, developing modular apps	Exposure:description, explanation,examples, discussion of case studies	
4. Fundamentals of graphical programming in G 3/3: sequential programming, state machines, parallelism, multiple loops architectures, global variables, event driven programming	Exposure:description, explanation,examples, discussion of case studies	
5. Programming Real Time Systems: introduction, components, devices configuration	Exposure:description, explanation,examples, discussion of case studies	
6. Architecture of Real Time Systems: Host/Target, multithreading, execution control, timing control, interprocess communication	Exposure:description, explanation,examples, discussion of case studies	
7. Optimization of Real Time Applications: requirements analysis, Target constraints, communication in distributed systems, memory management and system monitoring, realibility, debugging, testing, deploying	Exposure:description, explanation,examples, discussion of case studies	
8. PFGA Systems: components, compilation, timing issues, Single-Cycle Loop execution	Exposure:description, explanation,examples, discussion of case studies	
9. Image processing using VI: Introduction to Machine Vision, image acquisition and display, calibration, measurements. Image identification, bar codes, optical recognition of graphical characters	Exposure:description, explanation,examples, discussion of case studies	
10. Human Machine Interface, data output on mobile devices: graphical interfaces, G web server, LabView WebServices, SCADA elements with applications, Data Dashboard	Exposure:description, explanation,examples, discussion of case studies	
11. Rapid Prototyping. MyRIO; emebedded systems in education: MyRIO; Measurements applications; Simple control Applications; IIoT and Embedded Systems	Exposure:description, explanation,examples, discussion of case studies	
12. Real Time Systems Modelling: Model-in-the-loop, SW-in-the-loop, HW-in-the-loop	Exposure:description, explanation,examples, discussion of case studies	

Bibliography

1. Horia Hedesiu, Radu Munteanu jr. –Introducere in Programare Grafica Instrumentala, ISBN 973-9357-48 Mediamira, Cluj-Napoca, 2003
2. Gabriel Chindris, Horia Hedesiu - Proiectarea Grafica a Sistemelor de Control Pentru Aplicatii Industriale, ISBN 978-973-713-242-0, Editura Mediamira Cluj-Napoca, 2009
3. National Instruments Corp – LabVIEW Core 1 Course Manual, 2013 Edition
4. National Instruments Corp – LabVIEW Core 2 Course Manual, 2013 Edition
5. National Instruments Corp – LabVIEW Core 3 Course Manual, 2013 Edition
6. Kye-Si Kwon, Steven Ready - Practical Guide to Machine Vision Software: An Introduction with LabVIEW, Jan. 2015
7. Blume, Peter A. - The LabVIEW Style book, ISBN 0-13-145835-3, Pearson Education, 2007

8.2 Seminar / laboratory	Teaching methods	Remarks

1. Graphical programming in G 1/2	Explation, dialogue, case studies	
2. Graphical programming in G 2/2	Explation, dialogue, case studies	
3. Real Time Application development	Explation, dialogue, case studies	
4. Real Time Systems Architecture	Evaluation	
5. Image processing using VI	Explation, dialogue, case studies	
6. Rapid Prototyping	Explation, dialogue, case studies	
7. Final project turn-in	Evaluation	

Bibliography

1. Horia Hedesiu, Radu Munteanu jr. –Introducere in Programare Grafica Instrumentala, ISBN 973-9357-48 Mediamira, Cluj-Napoca, 2003
2. Gabriel Chindris, Horia Hedesiu - Proiectarea Grafica a Sistemelor de Control Pentru Aplicatii Industriale, ISBN 978-973-713-242-0, Editura Mediamira Cluj-Napoca, 2009
3. National Instruments Corp – LabVIEW Core 1 Course Manual, 2013 Edition
4. National Instruments Corp – LabVIEW Core 2 Course Manual, 2013 Edition
5. National Instruments Corp – LabVIEW Core 3 Course Manual, 2013 Edition
6. Kye-Si Kwon, Steven Ready - Practical Guide to Machine Vision Software: An Introduction with LabVIEW, Jan. 2015
7. Blume, Peter A. - The LabVIEW Style book, ISBN 0-13-145835-3, Pearson Education, 2007

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 programs of all major universities in Romania and abroad;
- The content of the course is considered by 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	Final project: architecture & design pattern application	Project grading	40%
10.5 Seminar/laboratory	Lab Assignments	-Mini-projects grading	60%
10.6 Minimum standard of performance			
- 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.			

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

Not applicable.

Date: ...	Signature of course coordinator Prof.Dipl.Eng.PhD. Horia HEDESIU	Signature of seminar coordinator Prof.Dipl.Eng.PhD. Horia HEDESIU

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*.".