Syllabus

1. Data about the study program

1.1 Institution	Technical University of Cluj-Napoca
1.2 Faculty	Automation and Computer Science
1.3 Departament	Automation
1.4 Field of study	Automation. Applied Informatics and Intelligent Systems
1.5 Cycle of study	Bachelor of Science
1.6 Program of study/Qualification	Automation and Applied Informatics (English)
1.7 Form of education	Full time

2. Data about the subject

2.1 Subject name	***Fundamentals of Computational Systems***					Discipline code	3.00
2.2 Course responsible/lecturer	Prof.dr.ing. Honoriu VĂLEAN – Honoriu@Valean@aut.utcluj.ro						
2.3 Teachers in charge of applications	Conf.dr.ing. Dan GOȚA - Dan.Gota@aut.utcluj.ro As.drd.ing Alexandru CIOBOTARIU – Alexandru.Ciobotariu@aut.utcluj.ro						
2.4 Year of study	1	2.5 Semester	1	2.6 Assessment *(E/C/V)*			E
2.7 Type of subject	*DF – fundamental, DS – specialty, DC – complementary*						DF
	DOB – compulsory, DOP – elective, DFA – optional						DOB

3. Estimated total time

3.1 Number of hours per week	4	of which:	3.2 Course	2	3.3 Seminar		3.3 Laboratory	2	3.3 Project	0
3.4 Total hours in the teaching plan	56	of which:	3.5 Course	28	3.6 Seminar		3.6 Laboratory	28	3.6 Project	0

3.7 Time distribution (hours per semester) for:	
(a) Manual, lecture material and notes, bibliography	28
(b) Supplementary study in the library, online and in the field	14
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays	14
(d) Tutoring	10
(e) Exams and tests	3
(f) Other activities:	28

3.8 Total hours individual study hours (sum of (3.3(a)...3.3(f)))	69
3.9 Total hours per semester (3.2+3.4)	125
3.10 Number of credits points	5

4. Pre-requisites (where appropriate)

4.1 Curriculum	-
4.2 Competence	-

5. Requirements (where appropriate)

5.1. For the course	-
5.2. For the applications	-

6. Specific competences

Professional competences	C1 The use of mathematics, physics, measurement techniques, technical graphics, mechanical, chemical, electrical and electronic engineering knowledge in automation, applied informatics and intelligent systems C1.1 Applying in professional communication of the concepts, theories and methods of fundamental sciences used in automation, applied informatics and intelligent systems. C1.3 Solving common problems in the field of automation, applied informatics and intelligent systems by identifying appropriate techniques, principles, methods and by applying mathematics, with an emphasis on numerical calculation methods.

Cross competences	

7. Expected learning outcomes

Knowledge	The student/graduate identifies and describes concepts, principles and methods of computer science; explains and interprets theoretical and experimental results in computer science; describes, identifies and summarizes fundamental concepts in computer science and information technology and their application in concrete problems; operates with fundamental concepts and methods in the field of communications; designs, implements, tests, uses and knows how to ensure the maintenance of systems with general and dedicated equipment, including computer networks or IoT, for automation and applied computer science applications;
Skills	The student/graduate operates with basic computer science concepts, principles and methods; solves problems in mathematics, physics and chemistry with applicability in engineering and validates the solution obtained; acquires and processes data, interprets theoretical and experimental results; designs solutions, respecting relevant standards, for engineering problems of medium complexity that meet the specified needs, respecting public health, safety, welfare, environmental, sustainability and economic factors, as well as other specific constraints; explains the issues to be solved, argues for solutions in systems engineering based on the principles of mathematics and the laws of physics and uses the basic components in the field and the techniques for measuring electrical and non-electrical quantities; uses programming languages, environments and technologies and specific tools (algorithms, schemes, models, protocols, etc.) in solving well-defined problems in systems engineering; evaluates and selects, adapts and extends computer-aided design (CAD) tools, hardware and dedicated software packages, for the implementation of complex systems connected to systems engineering; models at the system level and realizes numerical control systems, interfaces and systems based on microcontrollers and microprocessors, using integrated hardware-software design (codesign) and programming engineering; recognizes and implements complex architectures for intelligent autonomous systems; uses intermediate and high-level programming languages, including languages used for the implementation of intelligent systems; uses advanced technologies in the design and implementation of software applications;
Responsibilities and autonomy	The student/graduate applies the values of ethics and deontology of the engineering profession; practices logical reasoning, evaluation and self-evaluation in decision-making; shows a spirit of initiative and action to update professional, economic and organizational culture knowledge; has honorable, responsible, ethical behavior, in the spirit of the law to ensure the reputation of the profession; tracks personal progress in learning, seeks feedback and adopts strategies for continuous professional and personal development; responsibly applies engineering tools and platforms, ensuring system integrity and security, respecting good practices in configuration, testing and implementation; plans, executes and manages tasks autonomously, from problem definition to solution, demonstrating initiative in adopting new tools, methods and technologies.

8. Course objectives

General objective	understanding the structure, internal representation and operation of the computer
Specific objectives	understanding computer architecture understanding the internal representation of data understanding the role of an OS basic knowledge of Linux knowledge of using AI in the field of computing systems

9. Contents

9.1 Lecture	No.hours	Teaching methods	Notes
Evolution and architecture of computing systems: fundamental components, classification and emerging trends in processing	2		
Information representation and encoding in computing systems and artificial intelligence	2		
Linux Operating System Administration: Installation, Files, Users, and Bash Scripting	2		

Operating systems: fundamentals, historical evolution and modern classifications (embedded, distributed, real-time, mobile, desktop and cloud)	2	Teaching using laptop and projector, interactive course, debate	In case of force majeure, courses will be held online on the Teams platform.
Internal functionalities and management mechanisms in operating systems: kernel, processes, memory, files and AI services	2		
Advanced Models and Contemporary Trends in Operating System Design and Organization	2		
Parallel and Distributed Computing in the Cloud and Edge Era: The Role of AI in Data Center Optimization	2		
Network Fundamentals and Configuration: TCP/IP, Services, and AI Automation	2		
Operating System Security: AI-assisted Access Control, Encryption, Firewall, and Threat Detection Monitoring	2		
AI-powered OS resource monitoring, management and automation	2		
Intelligent Computing Systems: Scripting, Automation, and AI Integration	2		
Intelligent Systems: Artificial Intelligence, Hardware Infrastructure, and Cloud Services	2		
Integrating AI into Efficient Windows Network and System Management	2		
Bibliography 1. W. Kurt. Linux programming by example. Year: 2000 Cota 498.011 3 2. D.P. Bovet, M. Cesati. Understanding the Linux kernel Year: 2001 Cota 502.550 3. Any Linux book or documentation			
9.2 Aplications (seminar/laboratory/project/practical work)	No.hours	Teaching methods	Notes
Presentation of the fundamental components of computing systems	2	Presentation of examples, demonstrations, discussions, practical applications	In case of force majeure, courses will be held online on the Teams platform.
Bases of numeration	2		
Internal representation of information	2		
Installation of the Linux operating system, scripting	2		
Kernel, processes, memory, files and AI services	2		
System calls and SYSGEN	2		
Anomaly detection and maintenance	2		
Log analysis, services and alerts	2		
AI-assisted threat detection	2		
Persistent data space monitoring	2		
Automation scripts	2		
Cloud usage	2		
Windows and AI	2		
Laboratory Assessment	2		
Bibliography 1. W. Kurt. Linux programming by example. Year: 2000 Cota 498.011 3 2. D.P. Bovet, M. Cesati. Understanding the Linux kernel Year: 2001 Cota 502.550 3. Any Linux book or documentation			

10. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

The course is useful for any activity in the IT industry

11. Evaluation

Activity type	11.1 Assessment criteria	1.2 Assessment methods	11.3 Weight in the final grade
11.4 Course	Knowledge assessment, participation, course activity	Written or online exam on Teams	60%
11.5 seminar/laboratory/project/ practical work	Knowledge assessment, participation, laboratory activity	Practical or online on Teams exam	40%
11.6 Minimum performance standard Exam grade >= 5 and laboratory assessment grade >=5			

Date of filling in:		Title Firstname NAME	Signature
	Course	Prof.dr.ing. Honoriu VĂLEAN	
10.07.2026	Aplications	As.drd.ing. Alexandru CIOBOTARIU	

Date of approval by the Department Board	Head of Departament Prof.dr.ing. Honoriu VĂLEAN
Date of approval by the Faculty Council	Dean Prof.dr.ing. Vlad MUREȘAN