Universitatea Națională de Știință și Tehnologie Politehnica București

Facultatea de Electronică, Telecomunicații și

Tehnologia Informației

COURSE DESCRIPTION

1. Program identification information

1.1 Higher education institution	**National University of Science and Technology Politehnica Bucharest**
1.2 Faculty	**Electronics, Telecommunications and Information Technology**
1.3 Department	**Telecommunications**
1.4 Domain of studies	Electronic Engineering, Telecommunications and Information Technology
1.5 Cycle of studies	Bachelor/Undergraduate
1.6 Programme of studies	Networks and Telecommunications Software

2. Date despre disciplină

2.1 Course name (ro) (en)				Servicii de cloud și containerizare Cloud and Containerization Services			
2.2 Course Lecturer				Lect. PhD Cornelia Ionela BĂDOI			
2.3 Instructor for practical activities				Lect. PhD Cornelia Ionela BĂDOI			
2.4 Year of studies	4	2.5 Semester	2	2.6. Evaluation type	V	2.7 Course regime	Ob
2.8 Course type		S	2.9 Course code	04.S.08.O.313		2.10 Tipul de notare	Nota

3. Total estimated time (hours per semester for academic activities)

3.1 Number of hours per week	3	Out of which: 3.2 course	2	3.3 seminary/laboratory	1
3.4 Total hours in the curricula	42	Out of which: 3.5 course	28	3.6 seminary/laboratory	14
Distribution of time:					hours
Study according to the manual, course support, bibliography and hand notes Supplemental documentation (library, electronic access resources, in the field, etc) Preparation for practical activities, homework, essays, portfolios, etc.					52
Tutoring					2
Examinations					4
Other activities (if any):					0

3.7 Total hours of individual study	58.00
3.8 Total hours per semester	100
3.9 Number of ECTS credit points	4

4. Prerequisites (if applicable) (where applicable)

4.1 Curriculum	Completion of the following disciplines: Operating systems, Object-oriented programming, Networks and services.
4.2 Results of learning	General knowledge of Linux systems, programming languages (Python), networking.

5. Necessary conditions for the optimal development of teaching activities (where applicable)

5.1 Course	According to the university studies regulations at UNSTPB. The presentations are delivered using modern display technologies, such as a video projector or an LCD screen.
5.2 Seminary/ Laboratory/Project	The laboratory takes place on a dedicated infrastructure that includes computers or laptops with Internet access, running either a Linux operating system or a Linux virtual machine (Ubuntu). A variety of software tools are installed and used, including: • virtualization software (VirtualBox), • containerization software (Docker), • version control software (Git), • container orchestration software (Kubernetes), • software for automating application testing, delivery, and deployment processes (Jenkins).

6. General objective *(Reffering to the teachers' intentions for students and to what the students will be thought during the course. It offers an idea on the position of course in the scientific domain, as well as the role it has for the study programme. The course topics, the justification of including the course in the currcula of the study programme, etc. will be described in a general manner)*

This course is part of the field of Information Technology (IT), specifically within the subfield of virtualization and containerization. Its primary objective is to familiarize students with the main approaches, models, and explanatory theories of this subfield, applied to the development of practical applications and problem-solving, with direct relevance for enhancing student learning and aligning with emerging technologies used in industry.

In particular, the course addresses both fundamental and advanced topics, concepts, and principles, such as virtualization, containerization, system orchestration, and networking design elements for system interconnection. These are explored using dedicated tools, including ESXi, VirtualBox, Terraform, Docker, Kubernetes, Jenkins, as well as telecommunications protocols such as TCP/IP.

Through these components, the course aims to develop a solid foundation of theoretical knowledge and practical skills, enabling students not only to use but also to design systems based on virtualization and containerization, while at the same time providing them with a comprehensive overview of the essential methodological and procedural aspects of the field.

7. Competences *(Proven capacity to use knowledge, aptitudes and personal, social and/or methodological abilities in work or study situations and for personal and proffesional growth. They refflect the empolyers requirements.)*

Specific Competences	• Demonstrates solid knowledge, both fundamental and advanced, in the design and development of systems involving virtualization, containerization, orchestration, and computer network configuration through the use of virtual machines and containers. • Correlates and integrates the acquired knowledge to achieve a deeper understanding of the field. • Applies theoretical knowledge in real-world contexts, demonstrating operational skills. • Uses standardized methods and tools specific to the field to carry out the processes of assessment and diagnosis of a given situation, based on the identified/reported problems, and identifies appropriate solutions. • Argues and analyzes coherently and correctly the context of applying the fundamental knowledge of the field, using the key concepts of the discipline and the specific methodology. • Oral and written communication in Romanian: uses the scientific vocabulary specific to the field for effective communication, both written and oral. • Oral and written communication in a foreign language (English): demonstrates the understanding and use of the technical and specialized vocabulary, predominantly based on English terminology specific to the field, in both oral and written expression.
Transversal (General) Competences	• Works in a team and communicates efficiently, coordinating efforts with others to solve moderately complex problem situations. • Demonstrates autonomy and critical thinking: the ability to think scientifically, independently search for and analyze data, as well as to draw and present conclusions and identify solutions. • Exhibits analytical and synthetic reasoning: concisely presents acquired knowledge as a result of systematic analysis. • Adheres to principles of academic ethics: properly cites bibliographic sources in research and documentation activities. • Applies elements of emotional intelligence to appropriately manage social-emotional situations encountered in academic, professional, or real-life contexts, demonstrating self-control and objectivity in decision-making or under stress.

8. Learning outcomes *(Synthetic descriptions for what a student will be capable of doing or showing at the completion of a course. The learning outcomes reflect the student's acomplishments and to a lesser extent the teachers' intentions. The learning outcomes inform the students of what is expected from them with respect to performance and to obtain the desired grades and ECTS points. They are defined in concise terms, using verbs similar to the examples below and indicate what will be required for evaluation. The learning outcomes will be formulated so that the correlation with the competences defined in section 7 is highlighted.)*

Knowledge	*The result of knowledge aquisition through learning. The knowledge represents the totality of facts, priciples, theories and practices for a given work or study field. They can be theoretical and/or factual.* The development of hardware components used in computers (memory, processor, hard disk) and the advent of high-speed computer networks (Gbps) have significantly increased the level of parallel execution of multiple applications and operating systems on the same physical machine. Applications running on different machines may require incompatible resources, which has led to the need for containerization (i.e., creating a structure that encapsulates an application along with all its dependencies, thereby ensuring strict isolation from other applications). Public and private cloud systems rely on these technologies, providing mechanisms for the creation of computing and networking infrastructures for various fields of activity (public and private companies, healthcare institutions, educational organizations, governmental or non-governmental entities, etc.). In practice, most activities previously conducted at the premises of these entities can now be carried out exclusively in the Cloud or in a hybrid configuration, wherin some components are physical and others are virtualized in the Cloud. The pervasive adoption of new technologies across all areas of society necessitates specialized training in the following directions: • Virtualization - creating virtual machines using various virtualization infrastructures, such as VMWare and KVM. • Containerization - using Docker technology for application isolation and delivery. • Orchestration - managing and automating infrastructure using tools like Terraform, Ansible, and Kubernetes. • Scheduling activities and execution workflows (sets of chained activities/pipelines) using Jenkins. • Interconnection - implementing and configuring the TCP/IP protocol stack. • Designing and developing systems that integrate and interconnect all the aforementioned concepts, components, and subsystems.
Skills	*The capacity to apply the knowledge and use the know-how for completing tasks and solving problems. The skills are described as being cognitive (requiring the use of logical, intuitive and creative thinking) or practical (implying manual dexterity and the use of methods, materials, tools and intrumentation).* · Selects and organizes relevant information in a given context. · Critically applies specific principles for the design and use of systems involving virtualization, containerization, orchestration, and automated testing. · Works productively as part of a team. · Drafts a scientific text. · Experimentally verifies the identified solutions. · Solves practical applications. · Appropriately interprets causal relationships. · Analyzes and compares the concepts of virtualization, containerization, and orchestration. · Identifies solutions and develops plans or project proposals. · Formulates conclusions based on experiments conducted. · Justifies the solutions identified and the approaches taken to problem solving.

Responsability and autonomy	*The student's capacity to autonomously and responsably apply their knowledge and skills.* • Selects suitable bibliographic sources and analyzes them. • Adheres to academic ethics principles by correctly citing all utilized bibliographic references. • Demonstrates openness and adaptability to new learning contexts. • Engages in collaboration with peers and academic staff during educational activities. • Demonstrates autonomy in organizing their own learning context or in addressing a problem situation. • Displays social responsibility through active involvement in student life and academic community events. • Promotes and contributes new solutions in their area of expertise to enhance the quality of social life. • Recognizes the value of their engineering contributions in identifying viable and sustainable solutions to social and economic problems (social responsibility). • Applies ethical and professional principles in analyzing the technological impact of proposed solutions within their field of expertise. • Analyzes and capitalizes on business and entrepreneurial development opportunities in their area of specialization. • Demonstrates management skills for real-world situations (time management, collaboration versus conflict).

9. Teaching techniques *(Student centric techniques will be considered. The means for students to participate in defining their own study path, the identification of eventual fallbacks and the remedial measures that will be adopted in those cases will be described.)*

Based on the analysis of students' learning characteristics and their specific needs, the teaching process will explore both expository methods (lecture, presentation) and conversational-interactive methods, grounded in discovery-based learning models that facilitate both direct and indirect exploration of reality (experiment, demonstration, modeling), as well as action-based approaches such as exercises, practical activities, and problem-solving.

Lectures will be supported by PowerPoint presentations or various videos made available to students. Each class will begin with a review of previously covered chapters, emphasizing concepts studied in the last session. Presentations will incorporate images and diagrams to ensure that information is easily understood and assimilated.

The course covers both theoretical content and practical activities designed to support students in their learning efforts and to foster optimal collaboration and communicative relationships, within a learning environment conducive to discovery.

Particular emphasis will be placed on practicing active listening and assertive communication skills, as well as on feedback construction mechanisms, as means of behavioral regulation in various situations and as ways to adapt pedagogical approaches to the students' learning needs.

10. Contents

COURSE		
Chapter	**Content**	**No. hours**
1	Introduction	2
2	Python programming recap course	2
3	Virtualization and containerization. Docker	4

4	The TCP/IP protocol stack in virtualization and containerization	2
5	Version control systems. Local Git	4
6	Version control systems. Collaborative work with GitHub	4
7	Container orchestration using Kubernetes	1
8	Infrastructure orchestration using Terraform	1
9	Jenkins. Overview. Task scheduling and execution lines. Freestyle pipelines	6
10	Jenkins. Declarative pipelines	2
	Total:	28

Bibliography:

1. Note de curs – platforma Moodle
2. “Pro GIT” book, https://git-scm.com/book/en/v2
3. Documentation – Oracle VirtualBox, https://www.virtualbox.org/wiki/Documentation
4. Docker Docs, https://docs.docker.com/
5. Jenkins User Documentation, https://www.jenkins.io/doc/
6. A successful Git branching model » nvie.com, https://nvie.com/posts/a-successful-git-branching-model/
7. GitHub flow - GitHub Docs, https://docs.github.com/en/get-started/using-github/github-flow
8. What is the best Git branch strategy? | Git Best Practices, https://www.gitkraken.com/learn/git/best-practices/git-branch-strategy
9. What Are the Best Git Branching Strategies, https://www.abtasty.com/blog/git-branching-strategies/
10. Getting started with Pipelines, https://docs.cloudbees.com/docs/cloudbees-ci/latest/pipelines/
11. Kubernetes Documentation, https://kubernetes.io/docs/home/
12. Terraform Resources Overview – Examples & Best Practices, https://spacelift.io/blog/terraform-resources
13. 20 Terraform Best Practices to Improve Your TF Workflow, https://spacelift.io/blog/terraform-best-practices

LABORATORY

Crt. no.	Content	No. hours
1	VirtualBox installation and Linux (Ubuntu) virtual machine creation	1
2	Network configuration on a Linux (Ubuntu) virtual machine. Network setup with multiple server types and client-server connectivity verification	1
3	Local version control system usage (Git)	1
4	Remote version control system usage (GitHub). Collaborative work	2
5	Installation of containerization software Docker on virtual machines	1
6	Installation of application-specific containers for various types of applications	2
7	Configuration of a network that simultaneously integrates servers running on virtual machines and on containers. Verification of client-server connectivity	1
8	Jenkins installation and configuration	1
9	Jenkins. Creation of execution pipelines	2
10	Final laboratory colloquium - project defense and evaluation	2
	Total:	14

Bibliography:

1. Learn Git Branching, https://learngitbranching.js.org/
2. Curs Python, https://github.com/crchende/python/tree/master/curs
3. What is the best Git branch strategy?, https://www.gitkraken.com/learn/git/best-practices/git-branch-strategy
4. What Are the Best Git Branching Strategies, https://www.abtasty.com/blog/git-branching-strategies/
5. Build and ship software on a single, collaborative platform, https://github.com/
6. Aplicația sysinfo, https://github.com/crchende/sysinfo
7. Aplicația site distribuitor bazata pe Flask, https://github.com/crchende/site_distribuitor
8. Jenkins, https://www.jenkins.io/
9. Debian/Ubuntu, https://www.jenkins.io/doc/book/installing/linux/#debianubuntu
10. Installation of Java, https://www.jenkins.io/doc/book/installing/linux/#installation-of-java
11. Get started with Pipelines, https://docs.cloudbees.com/docs/cloudbees-ci/latest/pipelines/

11. Evaluation

Activity type	11.1 Evaluation criteria	11.2 Evaluation methods	11.3 Percentage of final grade
11.4 Course	Final evaluation: • knowledge of fundamental theoretical concepts; • knowledge of how to apply theory to specific problems in the field of virtualization and containerization; • comparison and evaluation of various techniques and methods used in practice.	Written test	20%
11.5 Seminary/laboratory/project	Practical application	Development of a practical application based on the concepts acquired during the course and laboratory sessions.	80%
11.6 Passing conditions			
• Fulfilling the obligations characteristic of laboratory activities (participating in the planned works). • Obtaining 50% of the total score.			

12. Corroborate the content of the course with the expectations of representatives of employers and representative professional associations in the field of the program, as well as with the current state of knowledge in the scientific field approached and practices in higher education institutions in the European Higher Education Area (EHEA)

The course aims to provide students with a solid foundation in building systems that leverage virtualization, containerization, orchestration, and computer network configuration using virtual machines and containers.

Its curriculum mirrors the structure and rigor of comparable programs offered by the National University of Science and Technology Politehnica Bucharest.

The gathered knowledge prepares future specialists for careers in virtualization and containerization, offering essential competencies for positions in IT&C companies as well as academic and research institutions.

Furthermore, the course emphasizes the development of practical skills, enabling graduates to effectively address real-world scenarios and thereby enhance their potential to contribute meaningfully to the improvement of the socio-economic environment.

Date	Course lecturer	Instructor(s) for practical activities
25.09.2025	Lect. PhD Cornelia Ionela BĂDOI	Lect. PhD Cornelia Ionela BĂDOI
Date of department approval	Head of department	
26.09.2025	Conf. Dr. Serban Georgica Obreja	
Date of approval in the Faculty Council	Dean	
26.09.2025	Prof. Dr. Mihnea Udrea	