Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti
Facultatea de Electronică, Telecomunicaţii şi
Tehnologia Informaţiei

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)				Tehnologii de programare în Internet			
2.2 Course Lecturer				S.l./Lect. Dr. Laurentiu BOICESCU			
2.3 Instructor for practical activities				S.l./Lect. Dr. Laurentiu BOICESCU			
2.4 Year of studies	3	2.5 Semester	2	2.6. Evaluation type	E	2.7 Course regime	Ob
2.8 Course type		D	2.9 Course code	04.D.06.O.312		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.					4
Tutoring					0
Examinations					4
Other activities (if any):					0

3.7 Total hours of individual study	8.00
3.8 Total hours per semester	50
3.9 Number of ECTS credit points	2

4. Prerequisites (if applicable) (where applicable)

4.1 Curriculum	Computer Programming Data Structures and Algorithms Object-Oriented Programming
4.2 Results of learning	General knowledge of programming Working with data structures, pointers, objects, classes Writing object-oriented programs

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

5.1 Course	In accordance with the university internal rules The lectures will take place in a room equipped with computer and video projector
5.2 Seminary/ Laboratory/Project	The laboratories will take place in a room equipped with video projector, computers with NetBeans IDE, web-server (Apache Tomcat), Database server (MySQL), virtual machine hypervisor (VirtualBox)

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)*

The course aims at the familiarization of students with widely used Internet programming technologies: desktop (TCP and UDP socket level communications), Web (HTTP server and client technologies), mobile (Android, Windows Phone, etc.), access to databases, using different programming and markup languages: HTML, CSS, JavaScript, Java, PHP, C#, XML, SQL.
Specific objectives include: completion of student training in software engineering, with a focus on Internet programming technologies, acquiring knowledge needed to design and implement software applications for the Internet, creating practical applications by using theoretical notions studied within the course.

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	C3. The application of basic knowledge, concepts and methods on the architecture of computer systems, microcontrollers, programming languages and techniques C3.4 Writing software in object-oriented languages, starting from requirements up to the development, debugging and analysis of the results. C4. Writing technical specifications, procurement, deployment and operation of fixed and mobile communications equipment, as well as planning, configuration and integration of telecommunication services and information security elements - skills for applying general knowledge of advanced software technologies (Java, Qt / C ++, .NET / C #, PHP, MySQL, XML, etc.) to design and implement communication systems and services (web, mobile, etc.) C6. The use of programming languages and specialized tools for software engineering in integrated telecommunications systems C6.1 Definition of methodologies, languages and software instruments used in the development of communications software. C6.4 Use of object-oriented techniques for analysing and modeling sontware systems C6.5 Programming of web and networking application elements
Transversal (General) Competences	CT1. The methodical analysis of the daily issues, identifying the problems for which well-known solutions are already available, thus accomplishing the professional tasks CT3. Accommodation to new technologies, personal and professional development, through continuous training using printed documentation, specialized software and digital resources in Romanian and, at least, one international language

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.* **main programming and markup languages used in the development of web platforms (Java, HTML, CSS, JavaScript, SQL)** **programming models based on layered communications and the TCP/IP stack** **software engineering methodologies (design, implementation and debugging of web platforms)** **architectural models used in the development of web platforms**
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).* Uses principles that are specific to the development of web applications and motivates the choices Works in a team Elaborates scientific text Verifies identified solutions through experiment Solves practical applications Analyses and compares web development technologies and libraries Identifies solutions and develops solution plans/projects Motivates the identified solutions
Responsability and autonomy	*The student's capacity to autonomously and responsably apply their knowledge and skills.* Respects academic ethics principles, correctly citing the bibliographic sources used. Is receptive to new learning contexts. Collaborates with peers and teachers in carrying out course activities Shows autonomy in managing the learning/problem solving situation/context Awareness of the value of the contribution to identifying a viable / sustainable solution to solve problems in social and economic life (social responsibility). Analyses and capitalizes on business opportunities in the specializes field. Demonstrates skills in managing real-life situations (time management, collaboration, 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.)*

Starting from the analysis of students' learning patterns and their specific needs, the teaching process will explore both presentational (lecture, presentation) and conversational-interactive teaching methods, based on discovery and exploration models (experimentation, demonstration, modeling), as well as using exercises, practical activities and problem solving.
Lectures based on Power Point presentations (that will be made available to students) will be used in the teaching activity. Each course will begin with a recap of the chapters already covered, with an emphasis on the concepts covered in the previous course. The first lectures will recap the object-oriented notions studied

during the previous semesters.
Presentations rely on images and diagrams, making the information of the lectures easy to understand and assimilate.
This course offers information and practical activities meant to aid th students in their efforts to learn and develop optimal collaborative an communicational relationship in a climate suitable to learning through discovery.
The ability for teamwork in solving different learning assignments will be encouraged.

10. Contents

COURSE		
Chapter	**Content**	**No. hours**
1	Introduction to Internet programming technologies -Introduction to communications networks -Layered communications models; communication protocols; the TCP / IP stack -Introduction to the Internet and most used technologies: DNS, HTTP	4
2	Introduction to desktop Java technologies -Introduction to the Java programming language; basic programming conventions -Java language keywords, data types, control structures -Object-oriented programming in Java, classes, polymorphism -Input and output streams -Socket-level programming	6
3	Server-side web programming technologies -The client-server model for web applications -The HTTP protocol -The Java Servlet technology -Persistence for web applications. Sessions and cookies -The JSP (Java ServerPages) technology -Advanced web programming. Request delegation and forwarding -The MV (Model-View) and MVC (Model-View-Controller) architectural design models -The SQL language. Databases. Practical applications -Integration of databases in web applications	12
4	Client-side web programming technologies -Introduction to client-side web technologies -The HTML and CSS languages. Design of graphic interfaces for web applications -The Javascript language. Dynamic web interfaces	6
	Total:	28
Bibliography:		

LABORATORY		
Crt. no.	**Content**	**No. hours**
1	Familiarization with the development environments -Familiarization with NetBeans IDE 6.1 -Java object-oriented programming recap	2

2	Socket level Java Programming -Threads -Sockets in Java	2
3	Programming Web services using Java Servlets -The Java Servlet technology -HTTP servlets	2
4	Programming Web applications using Java ServerPages -The JSP technology -The MV (Model-View) design pattern -The MVC (Model-View-Controller) design pattern	2
5	Storing web applications data in MySQL databases -The SQL language. MySQL syntax -Using MySQL in Java applications	2
6	Client-side Web programming technologies -The HTML, CSS and JavaScript languages -Integration of HTML interfaces in Java Web applications	2
7	Final lab examination	0
	Total:	14
Bibliography:		

11. Evaluation

Activity type	11.1 Evaluation criteria	11.2 Evaluation methods	11.3 Percentage of final grade
11.4 Course	- knowledge of essential theoretical notions; - knowledge of how to apply the theory to solve specific problems; - analysis of theoretical techniques and methods	Verification during the semester at dates fixed at the beginning of the semester. Final verification during the session.	40%

11.5 Seminary/laboratory/project	- knowledge of how programs are designed and development their design skills - demonstrating the functioning of an implemented program	Final laboratory test comprising a theoretical and a practical component, checking the way the student solves a practical problem (implementation, testing, operation).	30%
	Homework: - the ability to analyze and synthesize on the topic at hand - the ability to implement the practical application, as required - the ability to present and argue on the practical implementation	Final project presentation	30%
11.6 Passing conditions			
Obtaining 50% of the total score. Obtaining a 50% of the score related to activities during the semester.			

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)

Through their activity, students gain the ability to provide solutions to existing problems and propose ideas to improve the current situation in the field of software development, more specifically in the field of Internet software.
The development of the lecture relied on knowledge, theoretical an practical aspects as well as phenomena described by the specialized literature, own published research and the experience of the lecturers.
The course aims to develop the graduates' skills to design and implement new software for the Internet, as well as to analyse, debug and improve existing ones. Thus, graduates of the course can contribute to the economic environment, in the field of Internet applications.

Date	Course lecturer	Instructor(s) for practical activities
25.09.2025	S.l./Lect. Dr. Laurentiu BOICESCU	S.l./Lect. Dr. Laurentiu BOICESCU

Date of department approval — Head of department

Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti

Facultatea de Electronică, Telecomunicaţii şi

Tehnologia Informaţiei

26.09.2025

Conf. Dr. Serban Georgica Obreja



Date of approval in the Faculty Council

Dean

26.09.2025

Prof. Dr. Mihnea Udrea

