Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti

Facultatea de Electronică, Telecomunicaţii şi

Tehnologia Informaţiei

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	**Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti**
1.2 Facultatea	**Electronică, Telecomunicaţii şi Tehnologia Informaţiei**
1.3 Departamentul	**Telecomunicaţii**
1.4 Domeniul de studii	Inginerie Electronică, Telecomunicaţii şi Tehnologii Informaţionale
1.5 Ciclul de studii	Licenţă
1.6 Specializarea	Rețele și Software de Telecomunicaţii

2. Date despre disciplină

2.1 Denumirea disciplinei (ro) (en)				Servicii de cloud şi containerizare Cloud and Containerization Services			
2.2 Titularul activităţilor de curs				S.l./Lect. Dr. Cornelia Ionela BĂDOI			
2.3 Titularul activităţilor de seminar / laborator				S.l./Lect. Dr. Cornelia Ionela BĂDOI			
2.4 Anul de studiu	4	2.5 Semestrul	2	2.6. Tipul de evaluare	V	2.7 Regimul disciplinei	Ob
2.8 Tipul disciplinei		S	2.9 Codul disciplinei	04.S.08.O.313		2.10 Tipul de notare	Nota

3. Timpul total estimat (ore pe semestru al activităţilor didactice)

3.1 Număr de ore pe săptămână	3	Din care: 3.2 curs	2	3.3 seminar/laborator	1
3.4 Total ore din planul de învăţământ	42	Din care: 3.5 curs	28	3.6 seminar/laborator	14
Distribuţia fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie şi notiţe Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate Pregătire seminarii/ laboratoare/proiecte, teme, referate, portofolii şi eseuri					52
Tutorat					2
Examinări					4
Alte activităţi (dacă există):					0

3.7 Total ore studiu individual	58.00
3.8 Total ore pe semestru	100
3.9 Numărul de credite	4

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	Parcurgerea următoarelor discipline: Sisteme de operare, Programare obiect-orientată, Rețele și servicii.
4.2 de rezultate ale învăţării	Cunoştinţe generale despre sistemele de Linux, limbaje de programare (Python), reţelistică.

5. Condiţii necesare pentru desfăşurarea optimă a activităţilor didactice (acolo unde este cazul)

5.1 Curs	Conform regulamentului studiilor universitare în UNSTPB. Prezentările sunt susţinute utilizând tehnologii moderne de afişare, precum videoproiectorul sau ecranul LCD.
5.2 Seminar/ Laborator/Proiect	Laboratorul se desfăşoară pe o infrastructură specifică ce include calculatoare sau laptopuri cu acces la Internet, pe care rulează sistemul de operare Linux sau o maşină virtuală Linux (Ubuntu). Se instalează şi se utilizează diverse tipuri de software, printre care: • software de virtualizare (VirtualBox), • software pentru containerizare (Docker), • software pentru controlul versiunilor (Git), • software pentru orchestrarea containerelor (Kubernetes), • software pentru automatizarea proceselor de testare, livrare şi instalare a aplicaţiilor (Jenkins).

6. Obiectiv general *(Se referă la intenţiile profesorilor pentru studenţi, la ceea ce studenţii vor fi învăţaţi în timpul cursului. Oferă o orientare cu privire la locul cursului în cadrul domeniului ştiinţific abordat, precum şi la rolul pe care acesta îl are în cadrul specializării studiate. Vor fi descrise de o manieră generală tematicile abordate, justificarea includerii cursului în planul de învăţământ al specializării studiate etc.)*

Această disciplină se studiază în cadrul domeniului tehnologiei informaţiei (IT), mai exact în subdomeniul virtualizării şi containerizării. Disciplina îşi propune să familiarizeze studenţii cu principalele abordări, modele şi teorii explicative ale subdomeniului virtualizării şi containerizării, utilizate în rezolvarea de aplicaţii practice şi probleme, cu relevanţă pentru stimularea procesului de învăţare la studenţi si alinierea la noile tehnologii utilizate în industrie.

În particular, disciplina abordează ca tematică specifică următoarele noţiuni de bază/avansate, concepte şi principii specifice: virtualizare, containerizare, orchestrare sistemelor, elemente de design de reţelistică pentru interconectarea sistemelor, folosind unelte specifice cum ar fi: ESXi, VirtualBox, Terraform, Docker, Kubernetes, Jenkins, etc., şi protocoale de telecomunicaţii TCP/IP.

Prin aceste componente, disciplina urmăreşte formarea unui set solid de cunoştinţe teoretice şi abilităţi practice care să permită studenţilor atât utilizarea, cât şi proiectarea sistemelor bazate pe virtualizare şi containerizare, oferindu-le totodată o viziune de ansamblu asupra aspectelor metodologice şi procedurale esenţiale ale domeniului.

7. Competenţe *(Capacitatea dovedită de a utiliza cunoştinţe, aptitudini şi abilităţi personale, sociale şi/sau metodologice în situaţii de muncă sau de studiu şi pentru dezvoltarea profesională şi personală. Reflectă cerinţele angajatorilor.)*

Specifice	• Demonstrează cunoştinţe solide, atât fundamentale, cât şi avansate, în domeniul construirii de sisteme care implică virtualizare, containerizare, orchestrare şi configurare de reţele de calculatoare folosind maşini virtuale şi containere. • Corelează şi integrează cunoştinţele dobândite pentru o înţelegere aprofundată a domeniului. • Aplică practic cunoştinţele teoretice în contexte reale, demonstrând competenţe operative. • Utilizează metode şi instrumente standardizate, specifice domeniului, pentru realizarea procesului de evaluare şi diagnoză a unei situaţii, în funcţie de problemele identificate/raportate, şi identifică soluţii adecvate. • Argumentează şi analizează coerent şi corect contextul de aplicare a cunoştinţelor de bază ale domeniului, utilizând concepte cheie ale disciplinei şi metodologia specifică. • Comunicare orală şi în scris în limba română: utilizează vocabularul ştiinţific specific domeniului, în vederea comunicării eficiente, atât în scris, cât şi oral. • Comunicare orală şi în scris într-o limbă străină (engleză): demonstrează înţelegerea şi utilizarea vocabularului tehnic şi de specialitate, preponderent bazat pe limba engleză, specific domeniului, în exprimarea orală şi scrisă.
Transversale (generale)	• Lucrează în echipă şi comunică eficient, coordonându-şi eforturile cu ceilalţi pentru rezolvarea de situaţii problemă de complexitate medie. • Autonomie şi gândire critică: abilitatea de a gândi în termeni ştiinţifici, de a căuta şi analiza date în mod independent, precum şi de a desprinde şi prezenta concluzii şi a identifica soluţii. • Capacitate de analiză şi sinteză: prezintă în mod sintetic cunoştinţele dobândite, ca urmare a unui proces de analiză sistematică. • Respectă principiile de etică academică: în activitatea de documentare citează corect sursele bibliografice utilizate. • Pune în practică elemente de inteligenţă emoţională în gestionarea socio-emoţională adecvată a unor situaţii din viaţa reală/academică/profesională, demonstrând stăpânire de sine şi obiectivitate în luarea deciziilor sau în situaţii de stres.

8. Rezultatele învăţării *(Sunt enunţuri sintetice referitoare la ceea ce un student va fi capabil să facă sau să demonstreze la finalizarea unui curs. Rezultatele învăţării reflectă realizările studentului şi mai puţin intenţiile profesorului. Rezultatele învăţării informează studenţii despre ceea ce se aşteaptă de la ei din punct de vedere al performanţei, pentru a obţine notele şi creditele dorite. Sunt definite în termeni concreţi, folosind verbe similare exemplelor de mai jos şi indică ceea ce se va urmări prin evaluare. Rezultatele învăţării vor fi astfel redactate încât să fie evidenţiată clar relaţia faţă de competenţele definite la punctul 7.)*

Cunoştinţe	*Rezultatul asimilării de informaţii prin învăţare. Cunoştinţele reprezintă ansamblul de fapte, principii, teorii şi practici legate de un anumit domeniu de muncă sau de studiu. Pot fi teoretice şi/sau faptice.* Dezvoltarea componentelor hardware utilizate în calculatoare (memorie, procesor, hard disk) şi a reţelelor de calculatoare cu viteze mari (Gbps) a permis creşterea semnificativă a nivelului de execuţie paralelă a mai multor aplicaţii şi sisteme de operare pe aceeaşi maşină fizică. Aplicaţiile care rulează pe maşini diferite pot necesita resurse incompatibile, ceea ce a generat necesitatea containerizării, adică a creării unei structuri care încapsulează aplicaţia împreună cu toate dependenţele sale, oferind astfel o izolare fermă faţă de celelalte aplicaţii. Sistemele de cloud publice şi private se bazează pe aceste tehnologii, furnizând mecanisme pentru crearea infrastructurii de calcul şi interconectare / reţelistică pentru diverse domenii de activitate (companii publice şi private, instituţii de sănătate, instituţii de învăţământ, organizaţii guvernamentale sau non-guvernamentale etc.). Practic, majoritatea activităţilor desfăşurate anterior în sediile acestor entităţi pot fi acum derulate în Cloud sau într-o configuraţie hibridă, în care unele componente sunt fizice, iar altele sunt virtualizate în Cloud. Gradul ridicat de penetrare a noilor tehnologii în toate domeniile sociale impune o pregătire specifică privind următoarele direcţii: • Virtualizare - crearea maşinilor virtuale utilizând diverse infrastructuri de virtualizare, cum ar fi VMWare şi KVM. • Containerizare - utilizarea tehnologiei Docker pentru izolarea şi livrarea aplicaţiilor. • Orchestrare - managementul şi automatizarea infrastructurii utilizând instrumente precum Terraform, Ansible şi Kubernetes. • Programarea activităţilor şi a fluxurilor de execuţie (set de activităţi înlănţuite / pipeline-uri) utilizând Jenkins. • Interconectare - implementarea şi configurarea stivei de protocoale TCP/IP. • Designul şi dezvoltarea unor sisteme care integrează şi interconectează toate conceptele, componentele şi subsistemele menţionate anterior.
Aptitudini	*Capacitatea de a aplica cunoştinţe şi de a utiliza know-how pentru a duce la îndeplinire sarcini şi a rezolva probleme. Aptitudinile sunt descrise ca fiind cognitive (implicând utilizarea gândirii logice, intuitive şi creative) sau practice (implicând dexteritate manuală şi utilizarea de metode, materiale, unelte şi instrumente).* • Selectează şi grupează informaţii relevante într-un context dat. • Utilizează argumentat principii specifice pentru construirea şi utilizarea sistemelor care implică virtualizare, containerizare, orchestrare şi testare automată. • Lucrează productiv în echipă. • Elaborează un text ştiinţific. • Verifică experimental soluţiile identificate. • Rezolvă aplicaţii practice. • Interpretează adecvat relaţiile de cauzalitate. • Analizează şi compară conceptele de virtualizare, containerizare şi orchestrare. • Identifică soluţii şi elaborează planuri de rezolvare/proiecte. • Formulează concluzii în urma experimentelor realizate. • Argumentează soluţiile identificate şi modurile de rezolvare.

Responsabilitate și autonomie	*Capacitatea cursantului de a aplica în mod autonom și responsabil cunoștințele și aptitudinile sale.* • Selectează surse bibliografice potrivite şi le analizează. • Respectă principiile eticii academice, citând corect sursele bibliografice utilizate. • Demonstrează receptivitate faţă de contexte noi de învăţare. • Manifestă colaborare cu colegii şi cadrele didactice în desfăşurarea activităţilor didactice. • Demonstrează autonomie în organizarea situației/contextului de învățare sau a situației problemă de rezolvat. • Manifestă responsabilitate socială prin implicarea activă în viaţa socială studenţească şi în evenimentele comunităţii academice. • Promovează şi contribuie cu soluţii noi, aferente domeniului său de specialitate, pentru îmbunătăţirea calităţii vieţii sociale. • Conştientizează valoarea contribuţiei sale în domeniul ingineriei la identificarea de soluţii viabile şi sustenabile pentru rezolvarea problemelor din viaţa socială şi economică (responsabilitate socială). • Aplică principii de etică şi deontologie profesională în analiza impactului tehnologic al soluţiilor propuse asupra mediului înconjurător în domeniul său de specialitate. • Analizează şi valorifică oportunităţi de afaceri şi de dezvoltare antreprenorială în domeniul său de specialitate. • Demonstrează abilităţi de management al situaţiilor din viaţa reală (gestionarea timpului, colaborare versus conflict).

9. Metode de predare *(Se vor avea în vedere metode care să asigure predarea centrată pe student. Se va descrie modul în care se asigură participarea studenților la stabilirea propriului parcurs de învățare, cum se identifică eventualele rămâneri în urmă și ce măsuri remediale se adoptă în astfel de cazuri.)*

Pornind de la analiza caracteristicilor de învăţare ale studenţilor şi de la nevoile lor specifice, procesul de predare va explora metode atât expozitive (prelegerea, expunerea), cât şi conversative-interactive, bazate pe modele de învăţare prin descoperire facilitate prin explorarea directă şi indirectă a realităţii (experimentul, demonstraţia, modelarea), dar şi metode bazate pe acţiune, precum exerciţiul, activităţile practice şi rezolvarea de probleme.

În activitatea de predare vor fi utilizate prelegeri susţinute pe baza unor prezentări PowerPoint sau a diferitelor filmuleţe care vor fi puse la dispoziţia studenţilor. Fiecare curs va începe cu recapitularea capitolelor deja parcurse, cu accent asupra noţiunilor studiate la ultimul curs. Prezentările vor utiliza imagini şi scheme astfel încât informaţiile să fie uşor de înţeles şi asimilat.

Această disciplină acoperă informaţii şi activităţi practice menite să sprijine studenţii în eforturile lor de învăţare şi în dezvoltarea unor relaţii optime de colaborare şi comunicare, într-un climat favorabil învăţării prin descoperire.

Se va urmări exersarea abilităţilor de ascultare activă şi comunicare asertivă, precum şi a mecanismelor de construcţie a feedback-ului, ca modalităţi de reglare comportamentală în situaţii diverse şi de adaptare a demersului pedagogic la nevoile de învăţare ale studenţilor. De asemenea, se va exersa abilitatea de lucru în echipă pentru rezolvarea diferitelor sarcini de învăţare.

10. Conţinuturi

CURS		
Capitolul	**Conţinutul**	**Nr. ore**
1	Introducere	2

2	Curs recapitulativ limbaj programare. Python	2
3	Virtualizare și Containerizare. Docker	4
4	Stiva de protocoale TCP/IP în virtualizare și containerizare	2
5	Sisteme de versionare. Git local	4
6	Sisteme de versionare. Lucru colaborativ cu GitHub	4
7	Orchestrare containere folosind Kubernetes	1
8	Orchestrare infrastructură folosind Terraform	1
9	Jenkins. Prezentare generală. Programare activități și linii de execuție. Pipeline-uri Freestyle	6
10	Jenkins. Pipeline-uri declarative	2
	Total:	28

Bibliografie:

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

LABORATOR

Nr. crt.	Conţinutul	Nr. ore
1	Instalarea VirtualBox și crearea unei mașini virtuale Linux (Ubuntu)	1
2	Configurarea rețelei pe mașina virtuală Linux (Ubuntu). Configurarea rețelei cu mai multe tipuri de servere și verificarea conectivității client-server	1
3	Utilizarea unui sistem de versionare local (Git)	1
4	Utilizarea unui sistem de versionare remote (GitHub). Lucru colaborativ	2
5	Instalarea software-ului de containerizare Docker pe mașinile virtuale	1
6	Instalarea containerelor specifice pentru diverse tipuri de aplicații	2
7	Configurarea unei rețele care utilizează în paralel serverele de pe mașinile virtuale și de pe containere. Verificarea conectivității client-server	1
8	Instalare și configurare Jenkins	1
9	Jenkins. Crearea liniilor de execuție (pipelines)	2

10	Colocviu final de laborator - susţinere şi evaluare proiect	2
	Total:	14

Bibliografie:

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. Aplicaţia sysinfo, https://github.com/crchende/sysinfo
7. Aplicaţia 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. Evaluare

Tip activitate	11.1 Criterii de evaluare	11.2 Metode de evaluare	11.3 Pondere din nota finală
11.4 Curs	Evaluare finală: - cunoaşterea noţiunilor teoretice fundamentale; - cunoaşterea modului de aplicare a teoriei la probleme specifice din domeniul virtualizării şi containerizării; - compararea şi evaluarea diferitelor tehnici şi metode utilizate în practică.	Examen scris	20%
11.5 Seminar/laborator/proiect	Aplicaţie practică	Realizarea unei aplicaţii practice pe baza noţiunilor dobândite în cadrul cursului şi laboratorului.	80%
11.6 Condiţii de promovare			
• Îndeplinirea obligaţiilor caracteristice activităţilor de laborator (participarea la lucrările planificate). • Obţinerea a 50% din punctajul total.			

12. Coroborarea conţinutului disciplinei cu aşteptările reprezentanţilor angajatorilor şi asociaţiilor profesionale reprezentative din domeniul aferent programului, precum şi cu stadiul actual al cunoaşterii în domeniul ştiinţific abordat şi practicile în instituţii de învăţământ superior din Spaţiul European al Învăţământului Superior (SEÎS)

Subiectele cursului urmăresc ca studenţii să asimileze elementele de bază în domeniul construirii de sisteme care implică virtualizare, containerizare, orchestrare şi configurare de reţele de calculatoare folosind maşini virtuale şi containere.

Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti

Facultatea de Electronică, Telecomunicaţii şi

Tehnologia Informaţiei

Cursul are un conţinut similar cu cel al cursurilor desfăşurate de Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti.

Cunoştinţele transmise asigură pregătirea viitorilor specialişti în domeniile virtualizării şi containerizării, fiind utile tuturor celor care vor lucra în companiile IT&C sau în instituţiile academice şi de cercetare din domeniu.

Se urmăreşte dezvoltarea abilităţilor absolventului de a gestiona situaţii practice întâlnite în viaţa reală, în scopul creşterii contribuţiei acestuia la îmbunătăţirea mediului socio-economic.

Data completării	Titular de curs	Titular(i) de aplicaţii
21.09.2026	S.l./Lect. Dr. Cornelia Ionela BĂDOI	S.l./Lect. Dr. Cornelia Ionela BĂDOI

Data avizării în departament	Director de departament
22.09.2026	Conf. Dr. Serban Georgica Obreja

Data aprobării în Consiliul Facultăţii	Decan
22.09.2026	Prof. Dr. Mihnea Udrea