FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	**UNIVERSITATEA DIN ORADEA**
1.2 Facultatea	**Inginerie electrică și tehnologia informației**
1.3 Departamentul	**Calculatoare si tehnologia informației**
1.4 Domeniul de studii	**Calculatoare si tehnologia informației**
1.5 Ciclul de studii	**Licență**
1.6 Programul de studii/calificarea	**Tehnologia informației / inginer**

2. Date despre disciplină

2.1 Denumirea disciplinei	**TESTAREA SISTEMELOR DE CALCUL**						
2.2 Titularul activităţilor de curs	**Ş.l.dr.inf. Felicia Mirabela COSTEA**						
2.3 Titularul activităţilor de seminar /laborator/proiect	**Ş.l.dr.inf. Felicia Mirabela COSTEA**						
2.4 Anul de studiu	**IV**	2.5 Semestrul	**8**	2.6 Tipul de evaluare	**Ex**	2.7 Regimul disciplinei	**(O)**

(I) Impusă; (O) Opţională; (F) Facultativă

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 laborator	**2**
3.4 Total ore din planul de învăţământ	**56**	din care: 3.5 curs	**28**	3.6 laborator	**28**
Distribuţia fondului de timp					Ore
Studiul după manual, suport de curs, bibliografie şi notiţe					**7**
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren					**10**
Pregătire seminarii/laboratoare, teme, referate, portofolii şi eseuri					**10**
Tutoriat					**7**
Examinări					**10**
Alte activităţi.....					

3.7 Total ore studiu individual	**44**
3.9 Total ore pe semestru	**100**
3.10 Numărul de credite	**4**

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	(Conditionari)
4.2 de competențe	

5. Condiţii (acolo unde este cazul)

5.1. de desfăşurare a cursului	Cursul se desfășoară față în față. Cursul se defășoarăcu tehnicile disponibile: Laptop, Videoproiector, Tablă inteligentă
5.2. de desfăşurare a seminarului/laboratorului/proiectului	Prezența obligatorie la laborator 100% Laboratorul se desfășoară față în față. Recuperarea laboratoarelor se face conform Regulament privind activitatea profesională a studenților în baza Sistemului European de Credite Transferabile (ECTS) Nivelul maxim admis al absenţelor motivate sau nemotivate la activităţile practice (laboratoare, stagii, proiecte etc.) care pot fi recuperate este de până la 30% din totalul activităţilor. Lucrările delaborator se realizează utilizând mijloacele de lucru existente în laborator:

6.1. Competențele specifice acumulate	
Competențe profesionale	**C1.** Operarea cu fundamente științifice, inginerești și ale informaticii. **C2.** Proiectarea componentelor hardware, software și de comunicații. **C3.** Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor.

Competențe transversale	▪ Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputaţia profesiei
6.2. Rezultatele învățării	
Cunoștințe	▪ Studentul/absolventul cunoaște conceptele fundamentale ale testării sistemelor de calcul (eroare, defect, eșec, modele de defect software și hardware), nivelurile și tipurile de testare, criteriile de acoperire și modul de planificare și documentare a procesului de testare.
Aptitudini	▪ Studentul/absolventul proiectează cazuri de test prin tehnici funcționale și structurale, automatizează execuția testelor cu instrumente moderne (pytest, JUnit, Selenium/Playwright, Postman, JMeter) și integrează suitele de test în fluxuri de integrare și livrare continuă.
Responsabilitate și autonomie	▪ Studentul/absolventul demonstrează rigoare științifică și autonomie în elaborarea planurilor de testare, în raportarea și urmărirea defectelor și în interpretarea critică a metricilor de calitate (acoperire, densitate a defectelor, fiabilitate), respectând standardele și bunele practici din domeniu.

7. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

7.1 Obiectivul general al disciplinei	• Dezvoltarea capacității studenților de a planifica, proiecta și executa activități de testare pentru sisteme de calcul, în vederea verificării și validării corectitudinii, performanței, fiabilității și securității componentelor software și hardware. Cursul urmărește familiarizarea cu conceptele teoretice fundamentale ale testării și utilizarea instrumentelor software actuale pentru automatizarea testelor și integrarea acestora în procesele moderne de dezvoltare (Agile, DevOps, integrare și livrare continuă).
7.2 Obiectivele specifice	▪ Însușirea noțiunilor fundamentale de verificare și validare a sistemelor de calcul (modele de defect, niveluri și tipuri de testare, procesul de testare). ▪ Formarea abilității de a proiecta cazuri de test prin tehnici funcționale (clase de echivalență, valori limită, tabele de decizie, tranziții de stare) și structurale (acoperire de instrucțiuni, decizii, condiții și căi). ▪ Dezvoltarea competenței de a alege strategia de testare adecvată tipului de sistem (aplicații software, servicii web, sisteme distribuite, sisteme embedded). ▪ Dobândirea abilității de a utiliza instrumente actuale de testare și automatizare (pytest, JUnit, Selenium/Playwright, Postman, JMeter, SonarQube etc.) pentru implementarea și execuția suitelor de test. ▪ Formarea unei gândiri critice în analiza rezultatelor testării, în raportarea defectelor și în evaluarea metricilor de calitate și de acoperire. ▪ Dezvoltarea capacității de a realiza studii de caz și proiecte aplicative de testare care să reflecte probleme reale, relevante pentru piața muncii și pentru mediul de cercetare.

8. Conţinuturi

8.1 Curs	Metode de predare	Nr. ore / Observaţii
Introducere în testarea sistemelor de calcul • Rolul testării în asigurarea calității; verificare vs. validare • Terminologie: eroare, defect, eșec; modele de defect software și hardware • Procesul de testare: planificare, proiectare, execuție, raportare	Videoproiector, slide-uri şi predare interactivă la tablă	2
Niveluri și tipuri de testare • Testare unitară, de integrare, de sistem și de acceptanță		4

• Testare funcțională și nefuncțională; testarea de regresie • Poziționarea testării în modelele de dezvoltare: modelul V, Agile, abordarea shift-left		
Tehnici de testare statică • Inspecții, revizuiri de cod și analiză statică automată • Metrici de calitate a codului; instrumente de tip SonarQube și linters		2
Tehnici funcționale de proiectare a testelor (black-box) • Partiționarea în clase de echivalență, analiza valorilor limită, tabele de decizie • Testare bazată pe tranziții de stare, pe cazuri de utilizare și testare combinatorială (pairwise)		4
Tehnici structurale de proiectare a testelor (white-box) • Graful de flux al controlului; complexitate ciclomatică • Criterii de acoperire: instrucțiuni, decizii, condiții, căi; testarea prin mutații		4
Automatizarea testării și instrumente software moderne • Framework-uri xUnit (pytest, JUnit); duble de test: mock, stub, fake • Automatizarea testării interfețelor și a serviciilor web (Selenium/Playwright, Postman/REST)		4
Testarea nefuncțională a sistemelor de calcul • Testare de performanță, de încărcare și de stres; scalabilitate și fiabilitate • Testare de securitate: analiza vulnerabilităților, fuzzing, testare de penetrare		4
Testarea sistemelor hardware și embedded • Testabilitate, autotestare (BIST), interfața JTAG / boundary scan • Testarea sistemelor cu resurse limitate și a dispozitivelor IoT		2
Testarea continuă în fluxuri CI/CD. Aplicații practice și studii de caz.		2

Bibliografie:

1. Ammann, P., Offutt, J. (2016). Introduction to Software Testing (2nd Edition). Cambridge University Press.
2. Graham, D., Black, R., van Veenendaal, E. (2020). Foundations of Software Testing: ISTQB Certification (4th Edition). Cengage Learning.
3. Myers, G. J., Sandler, C., Badgett, T. (2011). The Art of Software Testing (3rd Edition). Wiley.
4. Crispin, L., Gregory, J. (2014). More Agile Testing: Learning Journeys for the Whole Team. Addison-Wesley.
5. Humble, J., Farley, D. (2010). Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation. Addison-Wesley.
6. Okken, B. (2022). Python Testing with pytest (2nd Edition). Pragmatic Bookshelf.
7. Molyneaux, I. (2014). The Art of Application Performance Testing (2nd Edition). O'Reilly Media.
8. Beyer, B., Jones, C., Petoff, J., Murphy, N. R. (2016). Site Reliability Engineering: How Google Runs Production Systems. O'Reilly Media.
9. Bushnell, M. L., Agrawal, V. D. (2000). Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits. Springer.
10. Wang, L.-T., Wu, C.-W., Wen, X. (2006). VLSI Test Principles and Architectures: Design for Testability. Morgan Kaufmann.

	Metode de predare	Nr. ore / Observații
8.3 Laborator	exemplificare & implementare	
Introducere în testarea automată cu **Python (pytest) și Java (JUnit): structura testelor unitare, aserțiuni, fixtures**		4
Proiectarea cazurilor de test prin tehnici funcționale: clase de echivalență, valori limită, tabele de decizie		4
Testare structurală: măsurarea acoperirii codului cu **coverage.py și JaCoCo; testare prin mutații**		4

Testarea de integrare a serviciilor web/REST: Postman, biblioteca requests, duble de test (mock, stub)		4
Automatizarea testării interfețelor utilizator cu **Selenium / Playwright**		4
Testarea de performanță și de securitate: **JMeter / Locust**, OWASP ZAP		4
Proiect final: plan de testare și suită automată de teste pentru un sistem la alegerea studentului, integrată într-un flux CI (GitHub Actions)	Prezentare proiect	4

Bibliografie:
1. pytest Documentation – https://docs.pytest.org
2. Selenium: www.selenium.dev – automatizarea testării aplicațiilor web.
3. Apache JMeter: jmeter.apache.org – testarea de performanță și de încărcare.
4. JUnit 5 User Guide – https://junit.org/junit5
5. OWASP Web Security Testing Guide – OWASP Foundation.
6. **Ammann, P., Offutt, J.** (2016). *Introduction to Software Testing (2nd Edition)*. Cambridge University Press.
7. **Graham, D., Black, R., van Veenendaal, E.** (2020). *Foundations of Software Testing: ISTQB Certification (4th Edition)*. Cengage Learning
.

9. Coroborarea conţinuturilor disciplinei cu aşteptările reprezentanţilor comunităţii epistemice, asociaţiilor profesionale şi angajatori reprezentativi din domeniul aferent programului

- **Relevanță academică**: tematica abordată (niveluri și tipuri de testare, tehnici funcționale și structurale, automatizarea testării, testarea nefuncțională) este aliniată la programele de studii similare din universități de prestigiu din țară și din străinătate.
- **Relevanță profesională**: competențele dobândite prin studiu și laborator (proiectarea cazurilor de test, automatizarea suitelor de test, analiza acoperirii și raportarea defectelor) sunt esențiale pentru domenii precum dezvoltarea software, asigurarea calității (QA), DevOps și securitatea cibernetică.
- **Cerințele angajatorilor**: cursul integrează instrumente software utilizate curent în industrie (pytest, JUnit, Selenium/Playwright, Postman, JMeter, SonarQube), dezvoltând abilități practice foarte căutate pe piața muncii, inclusiv pentru rolurile de test engineer și QA automation engineer.
- **Valoare adăugată**: studenții dobândesc cunoștințe teoretice și competențe aplicative care facilitează integrarea rapidă în echipe de dezvoltare și de asigurare a calității, unde testarea sistematică este cheia livrării unor produse fiabile.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din notafinală
10.4 Curs	**Nivelul de înțelegere teoretică** a conceptelor fundamentale ale testării sistemelor de calcul (niveluri și tipuri de testare, tehnici funcționale și structurale, criterii de acoperire). **Capacitatea de aplicare practică** a tehnicilor de proiectare a testelor pe studii de caz din domenii diferite (aplicații software, servicii web, sisteme embedded).	**scris** - 2/3 teorie si 1/3 probleme	**50 %**

10.6 Laborator	**Corectitudinea și claritatea cazurilor de test** proiectate și a soluțiilor implementate la temele teoretice și aplicative. **Calitatea proiectului**: plan de testare coerent, suită automată funcțională, acoperire relevantă a codului și raportare clară a defectelor.	**oral** – activitatea specifică fiecărei ședințe exerciții, întrebări	**50 %**
10.8 Standard minim de performanță			
nota scris minim 5 și nota proiect minim 5 Nota finală va fi calculată din media aritmetică a notelor scris și proiect.			

Data completării

02.09.2026

Titular curs

Ș.l.dr.inf.. CosteaMirabela
mira_costea@uoradea.ro

Titular laborator

Ș.l.dr.inf.. CosteaMirabela
mira_costea@uoradea.ro

Data avizării în departament
17.09.2026

Semnătura directorului de departament
Conf. dr. inf. Moisi Elisa
emoisi@uoradea.ro

Data avizării în consiliul facultății
18.09.2026

Semnătură Decan

Conf.dr.ing. Eugen GERGELY
egergely@uoradea.ro