

 UNIVERSITATEA TEHNICĂ A MOLDOVEI	FIȘA DISCIPLINEI/MODULULUI	Cod: F.O. 006	
	STRUCTURI DE DATE ȘI ALGORITMI	Ediția	1
		Revizia	1
		Pagina	1 / 4
 UNIVERSITATEA TEHNICĂ A MOLDOVEI	FIȘA DISCIPLINEI/MODULULUI	Cod: S.A. 004	
	Testarea software	Ediția	1
		Revizia	1
		Pagina	1 / 4



FIȘA DISCIPLINEI/ MODULULUI

MD-2045, CHIȘINĂU, STR. STUDENȚILOR, 9/7, TEL: 022 50-99-01 | FAX: 022 50-99-05, www.utm.md

TESTAREA SOFTWARE

1. Date despre disciplină/modul

Facultatea	Calculatoare, Informatică și Microelectronică				
Departamentul	Ingineria software și automatică				
Ciclul de studii	Studii superioare de licență, ciclul I				
Programul de studii	0613.3 Ingineria software				
Anul de studii	Semestrul	Tip de evaluare	Categoria formativă	Categoria de opționalitate	Credite ECTS
I (învățământ cu frecvență);	8	E	S – unitate de curs de specialitate	A - unitate de curs opțională	3

2. Timpul total estimat

Total ore în planul de învățământ	Din care				
	Ore auditoriale		Lucrul individual		
	Curs	Seminar	Studiul materialului teoretic	Pregătire aplicații	
90	30	15	30	15	

3. Precondiții de acces la disciplină/modul

Conform planului de învățământ	Programarea calculatoarelor, Programarea orientată pe obiecte, Bazele dezvoltării aplicațiilor, Baze de date, Analiza și modelarea sistemelor, Dezvoltarea aplicațiilor securizate, Tehnici și mecanisme de proiectare software. Proiectarea sistemelor informaționale.
Conform competențelor	Acționează creativ pentru a dezvolta aplicații și a selecta opțiunile tehnice adecvate. Participă la alte activități de dezvoltare. Optimizează dezvoltarea, întreținerea și performanța aplicațiilor prin utilizarea modelelor de design și prin reutilizarea soluțiilor existente.

4. Condiții de desfășurare a procesului educațional pentru

Curs	Pentru prezentarea materialului teoretic în sala de curs este nevoie de tablă, proiector și calculator.
Seminar	Studentii vor perfecta rapoarte conform condițiilor formulate. Termenul de prezentare a aplicațiilor practice – o săptămână după finalizarea acestora. Pentru prezentarea cu întârziere a lucrării, aceasta se depunează cu 1pct./săptămână de întârziere.

	FIȘA DISCIPLINEI/MODULULUI		Cod: F.O. 006	
	STRUCTURI DE DATE ȘI ALGORITMI		Ediția	1
			Revizia	1
			Pagina	2 / 4

5. Competențe specifice acumulate

Competențe profesionale	CP2 Integrarea componentelor – Ia în considerare propriile acțiuni și cele ale terților în procesul de integrare; – Respectă standardele și procedurile de control adecvate pentru a menține integritatea funcționalității și fiabilitatea generală a sistemului. CP3 Testarea aplicațiilor – Asigură expertiza pentru a supraveghea programele complexe de testare; – Asigură documentarea testelor și a rezultatelor pentru a furniza informații managerilor relevanți ai procesului/proceselor cum ar fi proiectanții, utilizatorii sau tehnicienii de întreținere; – Asigură conformitatea cu procedurile de testare, inclusiv trasabilitatea documentată; CP4 Implementarea soluțiilor – Ia în considerare propriile acțiuni și cele ale altora pentru a oferi soluții și a iniția o comunicare și o colaborare cu părțile interesate; – Asigură expertiza pentru a influența, prin consiliere și asistență, dezvoltarea de soluții.. CP8 Furnizarea de servicii – Stabilește programul sarcinilor operaționale; – Gestionează costurile și bugetul alocat în conformitate cu procedurile interne și constrângerile externe; – Identifică numărul optim de persoane necesare pentru a asigura gestionarea operațională a infrastructurii sistemului.
Competențe transversale	CT1. Autonomie și responsabilitate. Demonstrează executarea responsabilă a sarcinilor profesionale în condiții de autonomie CT2. Interacțiune socială. Execută rolurile și activitățile specifice muncii în echipă și distribuie sarcinile între membri pe niveluri subordonate CT3. Dezvoltare personală și profesională. Conștientizează nevoia de formare continuă cu utilizarea eficientă a resurselor și tehnicilor de învățare pentru dezvoltarea personală și profesională.

6. Obiectivele disciplinei/modulului

Obiectivul general	Dezvoltarea cunoștințelor fundamentale despre conceptele de testare software.
Obiectivele specifice	1. Învățarea unor tehnici și metodologii esențiale pentru testare manuală și automată. 2. Dezvoltarea abilității de a crea planuri de testare și de a evalua calitatea software-ului. 3. Utilizarea eficientă a instrumentelor moderne pentru testare și raportarea defectelor. 4. Formarea competențelor de asigurare a calității în ciclul de viață al dezvoltării software.

7. Conținutul disciplinei/modulului

	FIȘA DISCIPLINEI/MODULULUI	Cod: F.O. 006	
	STRUCTURI DE DATE ȘI ALGORITMI	Ediția	1
		Revizia	1
		Pagina	3 / 4
Tematica activităților didactice		Numărul de ore	
		învățământcu frecvență	
Tematica cursurilor			
T1 Introducere în testarea software: - Definiții și concepte fundamentale; - Importanța testării software în dezvoltarea modernă; - Tipuri de testare (funcțională, non-funcțională, regresivă, etc.); - Costurile și beneficiile testării.		4	
T2 Procese și metodologii de testare: - Testarea în cadrul ciclului de viață al dezvoltării software (SDLC); - Modele de dezvoltare software (Waterfall, Agile, DevOps) și impactul asupra testării; - Metodologii de testare: TDD (Test-Driven Development), BDD (Behavior-Driven Development).		6	
T3 Proiectarea testelor: - Principii de proiectare a testelor; - Tehnici de proiectare: testare bazată pe cerințe, cazuri de utilizare, tehnici bazate pe risc; - Strategii de testare (pozitivă, negativă, exploratorie).		12	
T4 Managementul defectelor: - Raportarea defectelor și ciclul de viață al unui defect; - Utilizarea instrumentelor de management (JIRA, Bugzilla); - Tehnici de prioritzare și triere a defectelor.		4	
T5 Asigurarea calității și metrice: - Diferența dintre testare și asigurarea calității; - Metrice în testare: acoperirea codului, densitatea defectelor, etc; - Optimizarea proceselor de testare.		4	
Total curs:		30	
Tematica seminarelor			
T1 Implementarea unui plan de testare pentru un proiect real		5	
T2 Integrarea testării manuale și automate		5	
T3 Analiza și raportarea defectelor		5	
Total curs:		15	

8. Referințe bibliografice

Principale	1. Marnie L. Hutcheson, <i>Software Testing Fundamentals: Methods and Metrics</i>, John Wiley & Sons, 2003, 408 p. 2. Lee Copeland, <i>A Practitioners Guide to Software Test Design</i>, Artech Hous, 2004, 294 p. 3. Ron Patton, <i>Software Testing</i>, Sams Publishing, 2005, 408 p. 4. Glenford J. Myers, <i>The Art of Software Testing</i>, John Wiley & Sons, 2004. 5. Pocatilu P. Costurile testării software. Editura ASE, București, 2004.
Suplimentare	6. Andrew Hunt, David Thomas. <i>Pragmatic Unit Testing in C#</i>. 7. Jeff Tian. <i>Software Quality Engineering</i>. John Wiley & Sons, 2005 8. Hung Q. Nguyen, <i>Testing Applications on the Web</i>, John Wiley & Sons, 2001

	FIȘA DISCIPLINEI/MODULULUI		Cod: F.O. 006	
	STRUCTURI DE DATE ȘI ALGORITMI		Ediția	1
			Revizia	1
			Pagina	4 / 4

9. Utilizarea IA generativă

Permise de utilizare	Utilizarea IA generative în cadrul temelor și proiectelor este permisă, cu condiția ca studenții să respecte următoarele reguli: - IA generativă poate fi utilizată pentru generarea de idei, structuri de text sau cod, dar toate materialele generate trebuie să fie revizuite și ajustate de către student pentru a se asigura că acestea corespund cerințelor academice. - Orice utilizare a IA generative trebuie să fie declarată în secțiunea de apendice a fiecărei lucrări, folosind fraza: "În timpul pregătirii acestei lucrări, autorul a utilizat [NUME INSTRUMENT / SERVICIU] în scopul [MOTIV]. După utilizarea acestui instrument/serviciu, autorul a revizuit și editat conținutul după cum a fost necesar și își asumă întreaga responsabilitate pentru conținutul lucrării."
Restricții de utilizare	Studenții nu trebuie să considere IA generativă ca o sursă de încredere pentru informații, deoarece nu oferă referințe clare sau surse documentate. - Nu este permisă citarea directă a conținutului generat de IA în lucrările academice ca și cum ar fi sursă primară. - Activitățile în care este interzis utilizarea IA generativă sunt specificare de profesor și sunt de regulă evaluări intermediare și finale sau care nu presupun activități de dezvoltare a competențelor profesionale.

10. Evaluare

Periodică		Curentă	Studiu individual	Examen
EP 1	EP 2			
15%	15%	5%	5%	60%
Standard minim de performanță Prezența la lecții; activitatea și calitatea pregătirii la / pentru prelegeri și lucrări de laborator; Obținerea notei minime de „5” la fiecare dintre atestări și lucrări de laborator; Obținerea notei minime de „5” la proiectul de an (se aplică după caz); Demonstrarea în lucrarea de examinare finală a cunoașterii tehnicilor de programare aplicate asupra structurilor de date prin implementarea algoritmilor studiați la rezolvarea problemelor în limbajul de programare C.				