

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Automatică și Calculatoare
1.3 Departamentul	Calculatoare
1.4 Domeniul de studii	Calculatoare și Tehnologia Informației
1.5 Ciclu de studii	Licență
1.6 Programul de studii / Calificarea	Tehnologia Informației / Inginer
1.7 Forma de învățământ	IF – învățământ cu frecvență

2. Date despre disciplină

2.1 Denumirea disciplinei	Sisteme de calitate în TI				Codul disciplinei	51
2.2 Titularul de curs	Șl. dr. Iordan Anca-Elena - Anca.lordan@cs.utcluj.ro					
2.3 Titularul activităților de seminar / laborator / proiect	-					
2.4 Anul de studiu	IV	2.5 Semestrul	7	2.6 Tipul de evaluare	C	
2.7 Regimul disciplinei	Categorii formative: <i>DF – fundamentală, DS - de specialitate, DC - complementară</i>					DS
	Opționalitate: <i>DOB - obligatorie, DOP - opțională, DFA - facultativă</i>					DOB

3. Timpul total estimat

3.1 Număr de ore pe săptămână	2	din care:	Curs	2	Seminar	-	Laborator	-	Proiect	-
3.2 Număr de ore pe semestru	28	din care:	Curs	28	Seminar	-	Laborator	-	Proiect	-
3.3 Distribuția fondului de timp (ore pe semestru) pentru:										
(a) Studiul după manual, suport de curs, bibliografie și notițe										13
(b) Documentare suplimentară în bibliotecă, pe platforme electronice de specialitate și pe teren										7
(c) Pregătire seminarii / laboratoare, teme, referate, portofolii și eseuri										-
(d) Tutoriat										-
(e) Examinări										2
(f) Alte activități:										-
3.4 Total ore studiu individual (suma (3.3(a)...)3.3(f)))				22						
3.5 Total ore pe semestru (3.2+3.4)				50						
3.6 Numărul de credite				2						

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Inginerie Software, Proiectare Software
4.2 de competențe	Șabloane de proiectare, Arhitecturi software

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Tabla, proiector, calculator.
5.2. de desfășurare a seminarului / laboratorului / proiectului	-

6. Competențele specifice acumulate

6.1 Competențe profesionale	• Analizează specificații software • Utilizează biblioteci de software • Utilizează șabloane de proiectare de software • Remediază erorile din software • Utilizează instrumente de inginerie software asistată de calculator
6.2 Competențe transversale	• Respectă reglementările • Gândește analitic • Interpretează informații matematice • Aplică cunoștințe științifice, tehnologice și ingineresti

7. Rezultatele așteptate ale învățării

Cunoștințe	Describe, identifică și sintetizează concepte, modele și metode aplicative specifice evaluării calității și fiabilității sistemelor software.
Aptitudini	Analizează, proiectează și implementează șabloane de testare din perspectiva calității sistemelor software. Utilizează metode și instrumente specifice estimării fiabilității software.
Responsabilitate și autonomie:	Demonstrează autonomie în aplicarea cunoștințelor și metodelor specifice Tehnologiei Informației, își asumă responsabilitatea pentru calitatea soluțiilor tehnice propuse și pentru respectarea standardelor profesionale, legale și etice, atât în activități individuale, cât și în cadrul echipelor de lucru.

8. Obiectivele disciplinei

8.1 Obiectivul general al disciplinei	Obiectivul cursului îl reprezintă familiarizarea cu tehnici moderne de evaluare a calității și fiabilității sistemelor software.
8.2 Obiectivele specifice	Pentru atingerea obiectivului general se urmăresc obiectivele specifice: ▪ Înțelegerea nevoii de calitate. ▪ Studiarea șabloanelor de testare. ▪ Analiza proiectelor software din punct de vedere al riscului de apariție a defectelor. ▪ Studiarea de modele pentru estimarea fiabilității software.

9. Conținuturi

9.1 Curs	Nr.ore	Metode de predare	Observații
Calitate software. Atribute de calitate	2	Expunere sistematică; Exemple; Discuții	
Estimarea costului unui produs software	2		
Managementul domeniului, timpului și riscului în software	2		
Metode Agile. Extreme Programming	2		
Activități de testare. Nivele și tipuri de testare	2		
Testare unitară. NUnit	2		
Dezvoltare dirijată de teste	2		
Șabloane pentru duble de testare. NSubstitute	2		
Șabloane de testare pentru arhitecturi stratificate	2		
Testare de integrare. Testare automată	2		
Fiabilitate software. Funcții specifice fiabilității software	2		
Metode de estimare și validare a indicatorilor de fiabilitate software	2		
Modele pentru estimarea fiabilității software	2		

Colocviu	2		
<i>Bibliografie (bibliografia minimală a disciplinei conținând cel puțin o lucrare bibliografică de referință a disciplinei, care există la dispoziția studenților într-un număr de exemplare corespunzător)</i> 1. C. Laporte, A. April, <i>Software Quality Assurance</i> , Wiley, 2018 2. J. Löwy, <i>Righting Software – A Method for System and Project Design</i> , Pearson Education, 2020 3. V. Khorikov, <i>Unit Testing: Principles, Practices, and Patterns</i> , Manning Publications, 2020 4. A. Tibi, <i>Pragmatic Test-Driven Development in C# and .NET</i> , Packt Publishing, 2022 5. G. Meszaros, <i>xUnit Test Patterns – Refactoring Test Code</i> , Addison Wesley, 2007 6. K. Beck, <i>Planning Extreme Programming</i> , Addison Wesley, 2000			
9.2 Aplicații - seminar / laborator / proiect	Nr.ore	Metode de predare	Observații
-			
<i>Bibliografie (bibliografia minimală pentru aplicații conținând cel puțin o lucrare bibliografică de referință a disciplinei care există la dispoziția studenților într-un număr de exemplare corespunzător)</i> -			

**Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.*

10. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului

Competențele dobândite sunt necesare pentru: A putea aborda un proiect din domeniul Tehnologiei Informației, începând din fazele premergătoare ale acestuia și până la încheierea dezvoltării lui. A testa și verifica corectitudinea unui software ce necesită cunoașterea metodelor și utilităților de testare și verificare existente.

11. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	Abilități de rezolvare a problemelor, activitate.	Evaluare scrisă	100%
Seminar	-	-	-
Laborator	-	-	-
Proiect	-	-	-
Standard minim de performanță: Condiții de promovare: Nota finală ≥ 5			

Data completării:	Titulari	Titlu, prenume / NUME	Semnătura
28.04.2026	Curs	Șl. dr. Anca-Elena IORDAN	
	Aplicații	-	

Data avizării în Consiliul Departamentului Calculatoare	Director Departament, Prof.dr.ing. Rodica Potolea
Data aprobării în Consiliul Facultății de Automatică și Calculatoare	Decan, Prof.dr.ing. Vlad Mureșan