

## 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 Ciclul de studii	Licență
1.6 Programul de studii / Calificarea	Tehnologia Informației / Inginer
1.7 Forma de învățământ	IF – învățământ cu frecvență
1.8 Codul disciplinei	51.00

### 2. Date despre disciplină

2.1 Denumirea disciplinei	<b>Sisteme de calitate in TI</b>				
2.2 Titularii de curs	Șl. dr. Anca-Elena Iordan - <a href="mailto:Anca.Iordan@cs.utcluj.ro">Anca.Iordan@cs.utcluj.ro</a>				
2.3 Titularul / Titularii activităților de Seminar / laborator / proiect	-				
2.4 Anul de studiu	IV	2.5 Semestrul	7	2.6 Tipul de evaluare ( E – examen, C – colocviu, V – verificare)	C
2.7 Regimul disciplinei	DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară				DS
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

### 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									0	
(d) Tutoriat									0	
(e) Examinări									2	
(f) Alte activități:									0	
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	<b>C5 - Întreținerea și exploatarea sistemelor hardware, software și de comunicații</b> <ul style="list-style-type: none"> <li>▪ <b>C5.1</b> - Identificarea și descrierea instrumentelor de modelare, simulare și evaluare a performanțelor sistemelor hardware, software și de comunicații</li> <li>▪ <b>C5.2</b> - Utilizarea unor cunoștințe interdisciplinare pentru asigurarea exploatării sistemelor hardware, software și de comunicații în raport cu cerințele domeniului de aplicații</li> <li>▪ <b>C5.3</b> - Utilizarea unor principii și metode de bază pentru asigurarea securității, siguranței și ușurinței în exploatarea sistemelor hardware, software și de comunicații</li> <li>▪ <b>C5.4</b> - Testarea și evaluarea calitativă a caracteristicilor funcționale și nefuncționale ale sistemelor informatice, pe baza unor criterii specifice</li> <li>▪ <b>C5.5</b> - Dezvoltarea de sisteme și aplicații pentru întreținerea și utilizarea de sisteme hardware, software și de comunicații</li> </ul>
6.2 Competențe transversale	<b>N/A</b>

## 7. Obiectivele disciplinei

7.1 Obiectivul general al disciplinei	Obiectivul cursului îl reprezintă familiarizarea cu tehnici moderne de evaluare a calității și fiabilității sistemelor software.
7.2 Obiectivele specifice	Pentru atingerea obiectivului general se urmăresc obiectivele specifice: <ul style="list-style-type: none"> <li>▪ Înțelegerea nevoii de calitate.</li> <li>▪ Studiarea metodelor de testare.</li> <li>▪ Analiza proiectelor software din punct de vedere al riscului de apariție a defectelor.</li> <li>▪ Studiarea de modele pentru fiabilitate software.</li> </ul>

## 8. Conținuturi

8.1 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. Managementul timpului	2		
Metode Agile. Extreme Programming	2		
Activități de testare. Nivele și tipuri de testare	2		
Unit testing. NUnit. NUnit Substitute	2		
TDD. ATDD	2		
Testare de integrare. Testare automată	2		
Fiabilitate și raportare a defectelor. Funcții specifice fiabilității	2		
Metode de estimare și validare a indicatorilor de fiabilitate	2		
Modele pentru fiabilitate software bazate pe timpul mediu de defectări	2		
Modele pentru fiabilitate software bazate pe numărarea defectelor	2		
Validarea fiabilității programelor	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> <ol style="list-style-type: none"> <li>1. C. Laporte, A. April, "Software Quality Assurance", Wiley, 2018.</li> <li>2. A. Tarlinder, "Developer Testing. Building quality into software", Addison Wesley, 2017.</li> <li>3. R. Pressmann, "Software Engineering – A Practitioner's Approach", McGraw Hill, 2019.</li> <li>4. I. Sommerville, "Software Engineering", Pearson, 2015.</li> <li>5. J. Tian, "Software Quality Engineering – Testing, Quality Assurance, and Quantifiable Improvement", Wiley, 2005.</li> <li>6. K. Beck, "Planning Extreme Programming", AddisonWesley, 2000.</li> <li>7. A. Adewole, "C# and .NET Core Test-Driven Development", Packt Publishing, 2018.</li> </ol>			
8.2 Aplicații (seminar/laborator/proiect)*	Nr.ore	Metode de predare	Observații

-			
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)			
-			

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

### 9. 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.
---

### 10. 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  $\geq 5$

Data completării:	Titulari	Titlu Prenume NUME	Semnătura
07.06.2024	Curs	Șl.dr.info. Anca-Elena Iordan	
	Aplicații	Șl.dr. info. Anca-Elena Iordan	
	-	-	

Data avizării în Consiliul Departamentului Calculatoare 20.02.2024	Director Departament, Prof.dr.ing. Rodica Potolea
Data aprobării în Consiliul Facultății de Automatică și Calculatoare 22.02.2024	Decan, Prof.dr.ing. Mihaela Dînsoreanu