

FIȘA DISCIPLINEI

1. Date despre program

Instituția de învățământ superior	Universitatea „Ștefan cel Mare” Suceava
Facultatea	Facultatea de Inginerie Electrică și Știința Calculatoarelor
Departamentul	Departamentul de Calculatoare
Domeniul de studii	Calculatoare și tehnologia informației
Ciclul de studii	Master
Programul de studii	Știința și ingineria calculatoarelor

2. Date despre disciplină

Denumirea disciplinei	INGINERIE SOFTWARE AVANSATĂ				
Titularul activităților de curs	Prof.dr.ing. Cristina TURCU				
Titularul activităților de seminar	as.dr.ing Ovidiu GHERMAN				
Anul de studiu	I	Semestrul	2	Tipul de evaluare	C
Regimul disciplinei	Categorია formativă a disciplinei DSI – Discipline de sinteză; DAP – Discipline de aprofundare				DAP
	Categorია de opționalitate a disciplinei: DI - impusă, DO - opțională, DF - facultativă				DI

3. Timpul total estimat (ore alocate activităților didactice)

I a) Număr de ore, pe săptămână	3	Curs	2	Seminar		Laborator	1	Proiect	
I.b) Totalul de ore (pe semestru) din planul de învățământ	42	Curs	28	Seminar		Laborator	14	Proiect	

II. Distribuția fondului de timp pe semestru	ore
II.a) Studiul după manual, suport de curs, bibliografie și notițe	35
II.b) Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	22
II.c) Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	26
II.d) Tutoriat	0
III. Examinări	2
IV. Alte activități:	0

Total ore studiu individual II (a+b+c+d)	83
Total ore pe semestru (Ib+II+III+IV)	127
Numărul de credite	5

4. Precondiții (acolo unde este cazul)

Curriculum	•
Competențe	•

5. Condiții (acolo unde este cazul)

Desfășurare a cursului	• PC, videoproiector și ecran, conexiune la Internet, note de curs, bibliografie recomandată	
Desfășurare aplicații	Seminar	•
	Laborator	• rețea de calculatoare cu minimum 12 stații conectate la Internet; • software: editor de texte specializat, framework-uri și biblioteci specifice; • lucrări de laborator; • videoproiector și ecran.
	Proiect	•

6. Competențe specifice acumulate

Competențe profesionale	CP5. Auditarea sistemelor și serviciilor informatice CP6. Abordarea, planificarea și finalizarea activităților de cercetare științifică în domeniul științei și ingineriei calculatoarelor
-------------------------	---

Competențe transversale	CT2. Managementul proiectelor complexe și utilizarea a diverse moduri de comunicare scrisă și orală
-------------------------	---

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

Obiectivul general al disciplinei	<ul style="list-style-type: none"> Cunoașterea conceptelor și problematicii ingineriei software avansate, în contextul dezvoltării programelor complexe, de calitate superioară.
-----------------------------------	---

8. **Conținuturi**

Curs	Nr. ore	Metode de predare	Observații
1. Introducere, principii de bază, consolidare noțiuni	4h	expunerea, prelegerea, conversație, studiu de caz, demonstrația	
1.1. Prezentare generală. Istoric. Importanța ingineriei software în 202x. Discuții.			
1.2. Concepte, terminologie. Recapitulare metodologii dezvoltare software.			
2. Cercetare în inginerie software	4h		
2.1. Prezentare generală. Direcții de cercetare			
2.2. Exemple de jurnale și conferințe științifice specifice			
2.3. Studii de caz, analize articole științifice relevante			
2. Inginerie software bazată pe componente	2h		
3. Inginerie software distribuită	2h		
4. Inginerie software orientată pe servicii	2h		
5. Inginerie software orientată pe cloud	2h		
6. Inginerie software pentru dispozitive mobile	2h		
8. Ingineria software în contextul IoT/IIoT/IoB	2h		
9. Securitatea în ciclul de viață software	2h		
10. Teorii și modele de acceptare ale noilor tehnologii	2h		
11. Software 2.0: inteligența artificială și ingineria software	2h		
13. Managementul calității software	2h		

Bibliografie

- Winters, Titus, Tom Manshreck, Hyrum Wright, and Safari, an O'Reilly Media Company. Software Engineering at Google. 1st ed. 2020.
- Ian Sommerville, *Engineering Software Products: An Introduction to Modern Software Engineering*, University of St Andrews, Scotland, 2019
- John Ousterhout, *A Philosophy of Software Design*, April, 2018
- Goeff Hulten's, *Building Intelligent Systems: A Guide to Machine Learning Engineering*, ISBN: 1484234316
- Michael Keeling, *Design It!: From Programmer to Software Architect (The Pragmatic Programmers)* 1st Edition, 2017
- Clean Architecture: A Craftsman's Guide to Software Structure and Design* (Robert C. Martin Series) 1st Edition, 2017
- I. Sommerville, *Software Engineering*, 10th Edition, Pearson India, 2017
- Eric J. Braude, Michael E. Bernstein, *Software Engineering: Modern Approaches*, Second Edition 2nd Edition, 2016
- Hans Van Vliet, *Software Engineering Principles and Practice*, Yded, 2015
- Roger S. Pressman, Bruce Maxim, *Software Engineering: A Practitioner's Approach* 8th Edition, 2014
- Software Engineering Body of Knowledge (SWEBOK) IEEE, 2004
- R.C. Martin, *Agile Software Development: Principles, Patterns, and Practices*, Prentice-Hall, 2002

13. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Grady Booch, Design Patterns: Elements of Reusable Object-Oriented Software 1st Edition, 1994
14. Articole științifice din reviste de specialitate, indexate WoS, BDI

Aplicații (proiect)	Nr. ore	Metode de predare	Observații
1. Elemente organizatorice: protecția muncii, prezentarea regulamentului intern, prezentarea platformelor de lucru și a uneltelor software specifice. Stabilirea echipelor de lucru și a temelor de cercetare per echipă.	2 h	lucrări practice, studii de caz, articole, proiecte, evaluare	
2. Analiza publicațiilor științifice naționale și internaționale, indexate în baze de date de prestigiu – Web of Science (WoS), IEEE Xplore (IEEE), ScienceDirect precum și a diverselor surse relevante în industrie – pentru identificarea direcțiilor de cercetare relevante pentru tema aleasă de echipă, în domeniul ingineriei software.	2 h		
3. Analiza publicațiilor științifice. Elaborare proiect de cercetare, conform normativelor și standardelor din domeniu cercetării.	2 h		
4. Elaborare proiect.	2 h		
5. Previzionare proiecte pe echipe (evaluare intermediară).	2 h		
6. Finalizare proiect.	2 h		
7. Evaluare proiect.	2 h		

Bibliografie

1. Articole științifice din baze de date indexate WoS, BDI
2. I. Sommerville, *Software Engineering*, 10th Edition, Pearson India, 2017
3. Gene Kim, Jez Humble, Patrick Debois, John Willis, John Allspaw, *The DevOps Book: How to Create World-Class Agility, reliability and Security in Technology Organisations*, IT Revolution Press, 2016.
4. Eric Brechner, *Agile Project Management with Kanban (Developer Best Practices)* 1st Edition, 2015
5. Roger S. Pressman, Bruce Maxim, *Software Engineering: A Practitioner's Approach* 8th Edition, 2014
6. Erik Klimczak, *Design for Software: A Playbok for Developers*, Wiley, 2013.
7. Len Bass, Paul Clements, Rick Kazman, *Software Architecture in Practice (SEI Series in Software Engineering)* 3rd Edition, 2012
8. Lisa Crispin, Janet Gregory, *Agile Testing: A Practical Guide for Testers and Agile Teams (1st Edition)*, Addison-Wesley Professional, 2009.
9. A. Cernian, A.D. Ioniță, *Noțiuni aplicative de inginerie a sistemelor de programe*, Ed. Matrixrom, București, 2009
10. I. Sommerville, *Software Engineering*, 8th Edition, 2008
11. Bocu Dorin, Bocu Razvan, *Modelare obiect orientată cu UML - Fundamentele modelării cu UML; inițiere în sabloane de proiectare utilizând sintaxa UML*, Ed. Albastra, Cluj-Napoca, 2006
12. B. Eckel, *Thinking in Patterns with Java*, <http://www.bruceeckel.com>, 2006
13. Software Engineering Body of Knowledge (SWEBOK) IEEE, 2004
14. R.C. Martin, *Agile Software Development: Principles, Patterns, and Practices*, Prentice-Hall, 2002

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

- Conținutul cursului și al laboratorului, prin problematica tratată, pune la dispoziția studentului cunoștințe conforme cu așteptările reprezentanților comunității epistemice și angajatorilor reprezentativi din domeniul calculatoarelor și tehnologiei informației. Tematica abordată se regăsește la universități de renume, din țară sau străinătate, cum ar fi, Universitatea Babeș-Bolyai, Facultatea de Matematică și Informatică; University of Washington Tacoma; Wright State University, Dayton, Ohio; Kent State University; Michigan State University; Universitat Politècnica de Catalunya- BarcelonaTech. Câteva exemple:

Compatibilitate națională:

- Universitatea Babeș-Bolyai Cluj-Napoca, Facultatea de Matematică și Informatică – masterat Inginerie Software – „Metodologii Agile de Dezvoltare a Aplicațiilor Software”: https://www.cs.ubbcluj.ro/files/curricula/2017/syllabus/IS_sem1_MME8143_en_tzutzu_2017_2808.pdf
- Universitatea Politehnică din București, Facultatea de Electronică, Telecomunicații și Tehnologia Informației, masterat Tehnologii software avansate pentru comunicații – „Inginerie Software pentru Telecomunicații”: http://www.telecom.pub.ro/TSAC/Fise_discipline_TSAC.pdf

Compatibilitate internațională:

- Universitatea Politehnică din Torino, masterat „Inginerie Informatică” - „Software Engineering” 04GSPOV: - https://didattica.polito.it/pls/portal30/gap.pkg_guide.viewGap?p_cod_ins=04GSPOV&p_a_acc=2019&p_header=S ;
https://didattica.polito.it/pls/portal30/gap.pkg_guide.viewGap?p_cod_ins=05BIDOV&p_a_acc=2019&p_header=S
- Universitatea din Manchester, masterat „Software Engineering”:
<https://www.manchester.ac.uk/study/masters/courses/list/08354/msc-acs-software-engineering/course-details/#course-profile>

10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	Participarea activă în timpul cursurilor	Evaluare continuă	10
	Însușirea cunoștințelor teoretice Cunoașterea terminologiei utilizate în domeniu și capacitatea de comunicare folosind limbaj de specialitate.	Evaluarea cunoștințelor din tematica studiată în timpul semestrului	40
Seminar			
Laborator			
Proiect	Demonstrarea capacității de analiză, sinteză, abstractizare și concretizare a cunoștințelor teoretice, în construirea unor argumentări, în identificarea unor probleme și a soluțiilor acestora.	Evaluarea proiectului elaborat în cadrul echipei	50
Standard minim de performanță			
<ul style="list-style-type: none"> • capacitatea de a folosi un limbaj adecvat într-o discuție pe teme de specialitate; • capacitatea de analiză și sinteză a lucrărilor științifice • capacitatea de a elabora studii de cercetare, articole științifice; • capacitatea de colaborare și lucru în echipe de cercetare; • cunoașterea a cel puțin 50% din cantitatea de informație vehiculată la orele de instruire 			

Data completării	Semnătura titularului de curs	Semnătura titularului de aplicație
21.09.2022		

Data avizării în departament	Semnătura directorului de departament
29.09.2022	

Data aprobării în Consiliul facultății	Semnătura decanului
30.09.2022	