

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	s. l. dr. ing. Felicia GIZA BELCIUG				
Anul de studiu	I	Semestrul	2	Tipul de evaluare	E
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	42
II.b) Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	80
II.c) Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	86
II.d) Tutoriat	0
III. Examinări	3
IV. Alte activități:	0

Total ore studiu individual II (a+b+c+d)	208
Total ore pe semestru (Ib+II+III+IV)	253
Numărul de credite	10

4. Precondiții (acolo unde este cazul)

Curriculum	•
Competențe	<ul style="list-style-type: none"> • C1. Operarea cu concepte și metode științifice în calculatoare și tehnologia informației • C4. Conceperea, proiectarea și implementarea rețelelor de calculatoare

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	<ul style="list-style-type: none"> • 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	C1. Operarea cu concepte și metode științifice avansate din calculatoare și tehnologia informației C2. Modelarea și rezolvarea problemelor cu tehnici de inteligență artificială în diverse domenii
-------------------------	--

	C4. Dezvoltarea și proiectarea sistemelor informatice complexe, paralele și distribuite C5. Auditarea sistemelor și serviciilor informatice C6. Abordarea, planificarea și finalizarea activităților de cercetare științifică în domeniul științei și ingineriei calculatoarelor
Competențe transversale	CT1. Comportarea onorabilă, responsabilă, etică, în spiritul legii, pentru a asigura reputația profesiei CT2. Preluarea diferitelor roluri în echipe de proiect și descrierea clară și concisă, verbală și în scris, în limba română și una internațională, a rezultatelor domeniilor de activitate CT3. Demonstrarea spiritului de creativitate, inițiativă și acțiune, pentru actualizarea cunoștințelor profesionale, economice și de cultură organizațională

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 actual
-----------------------------------	--

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. Motivația ingineriei software în 202x. Dicționar termeni. Recapitulare			
1.2. Exemple de jurnale și conferințe științifice specifice			
2. Cercetare în inginerie software	2h		
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		
7. Inginerie software pentru timp real	2h		
8. Reutilizare software. Re-inginerie software.	2h		
9. Securitatea în ciclul de viață software	2h		
10. DevOp	2h		
11. Inteligența artificială și ingineria software	2h		
13. Estimare cost software	2h		

Bibliografie

1. Ian Sommerville, *Engineering Software Products: An Introduction to Modern Software Engineering*, University of St Andrews, Scotland, 2019
2. John Ousterhout, *A Philosophy of Software Design*, April, 2018
3. Goeff Hulten's, *Building Intelligent Systems: A Guide to Machine Learning Engineering*, ISBN: 1484234316
4. Michael Keeling, *Design It!: From Programmer to Software Architect (The Pragmatic Programmers)* 1st Edition, 2017
5. *Clean Architecture: A Craftsman's Guide to Software Structure and Design* (Robert C. Martin Series) 1st Edition, 2017
6. I. Sommerville, *Software Engineering*, 10th Edition, Pearson India, 2017
7. Eric J. Braude, Michael E. Bernstein, *Software Engineering: Modern Approaches*, Second Edition 2nd Edition, 2016
8. Hans Van Vliet, *Software Engineering Principles and Practice*, Yded, 2015
9. Roger S. Pressman, Bruce Maxim, *Software Engineering: A Practitioner's Approach* 8th Edition, 2014

10. Software Engineering Body of Knowledge (SWEBOK) IEEE, 2004
11. R.C. Martin, *Agile Software Development: Principles, Patterns, and Practices*, Prentice-Hall, 2002
12. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Grady Booch, *Design Patterns: Elements of Reusable Object-Oriented Software* 1st Edition, 1994
13. Articole științifice din reviste de specialitate

Aplicații (laborator)	Nr. ore	Metode de predare	Observații
1. Protecția muncii. Prezentarea regulamentului intern. Analiză unelte software de lucru în echipă: sisteme de virtualizare, Git, servere web portabile, platforme Agile.	2 h	lucrări practice, studii de caz, articole, proiecte, evaluare	
2. Ingineria cerințelor. Modelarea cerințelor. Elaborarea documentului cerințelor utilizatorului.	2 h		
3. Managementul proiectului software. Managementul riscului. Diagrame Gantt. Diagrame PERT. Planificare etape proiect (Bitrix24, Redmine).	2 h		
4. Proiectare și implementare. Modelare diagrame UML.	2 h		
5. Testare automată. Blackbox testing – testarea automată a interfețelor cu SikuliX. Rapoarte de testare.	2 h		
6. Testare automată. Blackbox testing – Sikuli Robot Framework.	2 h		
7. Devops – platforme	2 h		

Bibliografie

1. I. Sommerville, *Software Engineering*, 10th Edition, Pearson India, 2017
2. Gene Kim, Jez Humble, Patrick Debois, John Willis, John Allspaw, *The DevOps Book: How to Create World-Class Agility, Reliability and Security in Technology Organizations*, IT Revolution Press, 2016.
3. Eric Brechner, *Agile Project Management with Kanban (Developer Best Practices)* 1st Edition, 2015
4. Roger S. Pressman, Bruce Maxim, *Software Engineering: A Practitioner's Approach* 8th Edition, 2014
5. Erik Klimczak, *Design for Software: A Playbook for Developers*, Wiley, 2013.
6. Len Bass, Paul Clements, Rick Kazman, *Software Architecture in Practice (SEI Series in Software Engineering)* 3rd Edition, 2012
7. Lisa Crispin, Janet Gregory, *Agile Testing: A Practical Guide for Testers and Agile Teams (1st Edition)*, Addison-Wesley Professional, 2009.
8. A. Cernian, A.D. Ioniță, *Noțiuni aplicative de inginerie a sistemelor de programe*, Ed. Matrixrom, București, 2009
9. I. Sommerville, *Software Engineering*, 8th Edition, 2008
10. 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
11. B. Eckel, *Thinking in Patterns with Java*, <http://www.bruceeckel.com>, 2006
12. Software Engineering Body of Knowledge (SWEBOK) IEEE, 2004
13. 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	<i>evaluare continuă</i>	10
	Însușirea cunoștințelor teoretice Cunoașterea terminologiei utilizate în domeniu și capacitatea de comunicare folosind limbaj de specialitate.	Evaluarea cunoștințelor teoretice și practice din tematica studiată în timpul semestrului	40
Seminar			
Laborator	Implicare în rezolvarea aplicațiilor practice de laborator. Demonstrarea capacității de analiză, sinteza, abstractizare și concretizare a cunoștințelor teoretice, în construirea unor argumentări, în identificarea unor probleme și a soluțiilor acestora.	<i>evaluare continuă</i> (prin metode orale și probe practice)	50
Proiect			
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 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
21.09.2020		

Data avizării în departament	Semnătura directorului de departament
30.09.2020	

Data aprobării în Consiliul facultății	Semnătura decanului
01.10.2020	