FIŞA DISCIPLINEI

1. Date despre program

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**				
Anul de studiu	I	Semestrul	2	Tipul de evaluare	C
Regimul disciplinei	Categoria formativă a disciplinei DF - fundamentală, DS - de specializare, DC – complementară				DS
	Categoria de opționalitate a disciplinei: DOB – obligatorie, DOP – opțională, DFA – facultativă				DOB

3. **Timpul total estimat** (ore alocate activităților didactice)

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

Distribuția fondului de timp pe semestru	ore
II. Studiu individual	80
III. Examinări	3
IV. Alte activități (precizați):	

Total ore studiu individual (II+III)	83
Total ore pe semestru (I.b+II+III+IV)	125
Numărul de credite	5

4. Competențe specifice acumulate

Competențe transversale	CT1. lucrează în echipe
Competențe profesionale	CP1. efectuează cercetare științifică CP4. concepe designul produsului CP5. asigura managementul de proiect CP8. dezvolta aplicații de procesare de date

5. Rezultatele învățării

Cunoștințe	Aptitudini	Responsabilitate și autonomie
Studentul/absolventul descrie, identifică, sumarizează concepte și metode avansate privitoare la modele și paradigme moderne de inginerie software și cunoaște rigorile științifice pentru elaborarea unei lucrări de cercetare, respectând normele etice.	Studentul / absolventul integrează cunoștințe din diferite domenii, identifică metode, modele și paradigme moderne de inginerie software aplicate în cercetarea și dezvoltarea sistemelor complexe, modulare și scalabile.	Studentul / absolventul poate derula managementul proiectelor complexe cu preluarea diverselor roluri în echipă, asumându-și gestionarea și transformarea situațiilor de muncă complexe și imprevizibile prin noi abordări strategice adecvate.

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

Obiectivul general al disciplinei	• Dezvoltarea competențelor avansate în ingineria software, utilizând inteligența artificială, în contextul dezvoltării programelor complexe, de înaltă calitate, care să îndeplinească cerințele clienților legate de funcționalitate, costuri și termene de livrare.

7. Conținutul predării și învățării

Curs	Nr. ore	Metode de predare	Observații
1. Introducere, principii de bază, consolidare noțiuni	**2h**	expunerea, prelegerea, conversație, studiu de caz, demonstrația	
1.1. Prezentare generală. Istoric. Importanța ingineriei software în 202x. Discuții.			
1.2. Concepte fundamentale, terminologie.			
2. Cercetare în inginerie software	**2h**		
2.1. Direcții de cercetare			
2.2. Exemple de jurnale și conferințe științifice specifice			
2.3. Metodologia cercetării în inginerie software			
2.4. Studii de caz, analize articole științifice relevante			
3. Managementul proiectelor software complexe	**2h**		
4. Ingineria avansată a cerințelor	**2h**		
5. Arhitecturi software avansate	**2h**		
6. Design avansat și calitate internă a codului	**2h**		
7. Inteligența artificială pentru ingineria software 7.1. Concepte de inteligență artificială relevante pentru ingineria software	**4h**		
7.2. Ingineria software asistată de inteligența artificială			
7.3. Studii de caz, analiza articolelor științifice relevante			
8. Inginerie software pentru IA	2h		
9. Ingineria calității software	**2h**		
10. Testarea software avansată	**2h**		
11. Securitatea în ciclul de viață software	**2h**		
12. Inginerie software pentru aplicații de timp real, IoT și edge	**2h**		
13. Etică profesională, inginerie software responsabilă și provocări emergente	**2h**		

Bibliografie minimală recomandată

- Stray, V., Hanssen, G. K., Barbala, A., Šmite, D., & Stol, K. J. (2025). What is Generative AI good for? Introduction to the special issue on Generative AI in software engineering. Information and Software Technology, 107857. https://www.sciencedirect.com/science/article/abs/pii/S095058492500196X
- Schieferdecker, Ina K. "Next-Gen Software Engineering: AI-Assisted Big Models." arXiv, September 26, 2024. https://doi.org/10.48550/arXiv.2409.18048.
- Bazzan, T., Olojo, B., Majda, P., Kelly, T., Yilmaz, M., Marks, G., & Clarke, P. M. (2024, September). Analysing the Role of Generative AI in Software Engineering-Results from an MLR. In European Conference on Software Process Improvement (pp. 163-180). Cham: Springer Nature Switzerland. https://link.springer.com/chapter/10.1007/978-3-031-71139-8_11
- Huang, Yuan, Yinan Chen, Xiangping Chen, Junqi Chen, Rui Peng, Zhicao Tang, Jinbo Huang, Furen Xu, and Zibin Zheng. "Generative Software Engineering." arXiv, April 3, 2024. https://doi.org/10.48550/arXiv.2403.02583.
- Bourque, P.; Fairley, R.E. Guide to the Software Engineering Body of Knowledge (SWEBOK) V4; IEEE Computer Society, 2024; ISBN 978-0-7695-5166-1. https://www.computer.org/education/bodies-of-knowledge/software-engineering/v4

- Lara Letaw. Handbook of Software Engineering Methods – 2nd ed., Oregon State University open textbook, 2024. https://open.umn.edu/opentextbooks/textbooks/1557.
- Fan, Angela, Beliz Gokkaya, Mark Harman, Mitya Lyubarskiy, Shubho Sengupta, Shin Yoo, and Jie M. Zhang. "Large Language Models for Software Engineering: Survey and Open Problems." In *2023 IEEE/ACM International Conference on Software Engineering: Future of Software Engineering (ICSE-FoSE)*, 31–53, 2023. https://doi.org/10.1109/ICSE-FoSE59343.2023.00008.
- Nguyen-Duc, Anh, Beatriz Cabrero-Daniel, Adam Przybylek, Chetan Arora, Dron Khanna, Tomas Herda, Usman Rafiq, et al. "Generative Artificial Intelligence for Software Engineering -- A Research Agenda." arXiv, October 28, 2023. https://doi.org/10.48550/arXiv.2310.18648.
- Sommerville, I. Software Engineering; 10th edition.; Pearson: Boston, 2016; ISBN 978-0-13-394303-0.
- Articole științifice din reviste de specialitate, indexate WoS, BDI.

Aplicații (~~Seminar~~ / ~~laborator~~ / ~~lucrări practice~~ / 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	Lucrări practice, studii de caz, articole, proiecte, evaluare.	
2. Managementul proiectului. Tipuri de diagrame (Gantt, Kanban etc.). Utilizarea platformelor software dedicate. Ingineria cerințelor. Elaborarea documentului specificațiilor.	2		
3. Modelarea UML a proiectului software. Gestionarea riscurilor. Analiza pattern-urilor de design.	2		
4. Elaborare raport de cercetare. Utilizarea platformelor software dedicate. Previzionare.	2		
5. Elaborare raport de cercetare. Refactorizare și aplicarea principiilor SOLID. Documentare.	2		
6. Elaborare raport de cercetare. Testare. Evaluarea calității. Plan de mentenanță.	2		
7. Evaluare raport de cercetare.	2		

Bibliografie minimală recomandată

- Bourque, P.; Fairley, R.E.. Guide to the Software Engineering Body of Knowledge (SWEBOK) V4 beta; IEEE
- Computer Society, 2024; ISBN 978-0-7695-5166-1. https://www.computer.org/education/bodies-ofknowledge/software-engineering.
- Lara Letaw. Handbook of Software Engineering Methods – 2nd ed., Oregon State University open textbook, 2024. https://open.umn.edu/opentextbooks/textbooks/1557.
- Visual Paradigm team. Learning Guides, 2024. https://www.visual-paradigm.com/guide/.
- Karlovs-Karlovskis, Uldis. "Generative Artificial Intelligence Use in Optimising Software Engineering Process: A systematic Literature Review." Applied Computer Systems 29.1. 2024: 68-77. https://doi.org/10.2478/acss-2024-
- 0009.
- Huang, Yuan, Yinan Chen, Xiangping Chen, Junqi Chen, Rui Peng, Zhicao Tang, Jinbo Huang, Furen Xu, and Zibin
- Zheng. "Generative Software Engineering." arXiv, April 3, 2024. https://doi.org/10.48550/arXiv.2403.02583.
- David J. Anderson, Andy Carmichael. "Essential Kanban Condensed", Lean-Kanban University, ISBN 978-0984521425, 2016. https://kanbanbooks.com/free-kanban-book-downloads/.
- Articole științifice din reviste de specialitate, indexate WoS, BDI.
- Ovidiu Gherman - Îndrumar de laborator ISA, versiune electronică, 2026.

8. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Evaluare finală	Însușirea cunoștințelor teoretice. Cunoașterea terminologiei utilizate în domeniu și capacitatea de comunicare folosind limbaj de specialitate.	Evaluarea prin test grilă a cunoștințelor din tematica studiată în timpul semestrului.	20%
	Cunoașterea termenilor specifici domeniului și utilizarea acestora în cadrul proiectului.	Evaluarea finală a proiectului elaborat.	40%
Evaluare finală	Demonstrarea capacității de analiză, sinteză, abstractizare și concretizare a cunoștințelor teoretice și practice, în construirea unor	Evaluarea temelor elaborate pe parcursul semestrului.	40%

	argumentări, în identificarea unor probleme și a soluțiilor acestora.		

Fișa disciplinei include, dacă este cazul, elemente adaptate persoanelor cu dizabilități, în funcție de tipul și gradul acestora.

Data completării	Grad didactic, nume, prenume, semnătura titularului de curs	Grad didactic, nume, prenume, semnătura titularului de aplicație
23.09.2026	prof.dr.ing. Cristina Elena TURCU	ș.l.dr.ing Ovidiu GHERMAN

Data avizării	Grad didactic, nume, prenume, semnătura responsabilului de program
24.09.2026	prof. dr. ing. Stefan-Gheorghe PENTIUC

Data avizării în departament	Grad didactic, nume, prenume, semnătura directorului de departament
24.09.2026	Prof. dr. ing. Ovidiu-Andrei SCHIPOR

Data aprobării în consiliul facultății	Grad didactic, nume, prenume, semnătura decanului
25.09.2026	Prof. dr. ing. Laurentiu- Dan MILICI