FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	Universitatea Politehnica Timișoara
1.2 Facultatea[1] / Departamentul[2]	Robotică, Inteligență Artificială și Sisteme Autonome / Calculatoare și Tehnologia Informației
1.3 Domeniul de studii (denumire/cod [3])	Informatică / 40
1.4 Ciclul de studii	Licență
1.5 Programul de studii (denumire/cod/calificarea)	Informatică / 011 / Informatică

2. Date despre disciplină

2.1a Denumirea disciplinei/Categoria formativă[4]			Dezvoltarea Software Asistată de AI / f				
2.1b Denumirea disciplinei în limba engleză			AI-Assisted Software Development				
2.2 Titularul activităţilor de curs			Ș.l.univ.dr.ing. Cătălin IAPĂ				
2.3 Titularul activităţilor aplicative[5]			Ș.l.univ.dr.ing. Cătălin IAPĂ				
2.4 Anul de studii[6]	3	**2.5** Semestrul	1	**2.6** Tipul de evaluare	E	**2.7** Regimul disciplinei[7]	DF

3. Timp total estimat - ore pe semestru: activități didactice directe (asistate integral sau asistate parțial) și activități de pregătire individuală (neasistate) [8]

3.1 Număr de ore asistate integral/săptămână	2, format din:	**3.2** ore curs	2	**3.3** ore seminar/laborator/proiect	0
3.1* Număr total de ore asistate integral/sem.	28, format din:	**3.2*** ore curs	28	**3.3*** ore seminar/laborator/proiect	0
3.4 Număr de ore asistate parțial/săptămână	, format din:	**3.5** ore practică		**3.6** ore elaborare proiect de diplomă	
3.4* Număr total de ore asistate parțial/ semestru	, format din:	**3.5*** ore practică		**3.6*** ore elaborare proiect de diplomă	
3.7 Număr de ore activități neasistate/ săptămână	1.6, format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			0.5
		ore studiu individual după manual, suport de curs, bibliografie și notițe			0.5
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			0.6
3.7* Număr total de ore activități neasistate/ semestru	22, format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			7
		ore studiu individual după manual, suport de curs, bibliografie și notițe			7
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			8
3.8 Total ore/săptămână [9]	3.6				
3.8* Total ore/semestru	50				
3.9 Număr de credite	2				

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	• Cunoștințe de bază de programare (Python/Java/C), noțiuni introductive de inteligență artificială.
4.2 de rezultatele învățării	• Abilitatea de a scrie și înțelege programe, gândire algoritmică.

5. Condiţii (acolo unde este cazul)

5.1 de desfăşurare a cursului	• Sală cu proiector, acces internet.
5.2 de desfăşurare a activităţilor practice	• Laborator cu calculatoare, conexiune internet, medii de dezvoltare (VS Code, Jupyter, Replit/Cursor, API-uri LLM).

6. Rezultatele învățării la formarea cărora contribuie disciplina

Cunoștințe	• C1. Studentul identifică, explică și argumentează concepte fundamentale de programare, algoritmi, logica si structuri discrete, inclusiv structuri de date.
Abilități	• A1. Studentul elaborează, dezvoltă și demonstrează soluții informatice complexe utilizând concepte moderne de programare, algoritmi eficienți și structuri de date avansate.
Responsabilitate și autonomie	• RA1. Studentul elaborează aplicații informatice pe baza unor decizii responsabile legate de optimizarea și integrarea acestora in medii socio-industriale diverse.

7. Obiectivele disciplinei (asociate rezultatelor învățării de la punctul 6)

- Formarea competențelor de utilizare eficientă și responsabilă a inteligenței artificiale în procesul de programare, prin înțelegerea tehnicilor de context engineering și integrarea lor în dezvoltarea software.
-

8. Conţinuturi[10]

8.1 Curs	Număr de ore	Metode de predare[11]
Introducere: AI în programare. Noțiunea de vibe coding. Exemple practice.	2	Prelegere susținută de material proiectat, explicații, discuții
Modele de limbaj mari (LLM) pentru cod: GPT, Codex, Code Llama.	2	
Prompt engineering: zero-shot, few-shot, chain-of-thought.	2	
Context engineering: definiție, bune practici, studii de caz. Retrieval Augmented Generation (RAG)	2	
Pair programming cu AI: rolul omului ca supervisor. Agentic AI in programare. Fișiere de cunoștințe pentru tooluri AI. Model Context Protocol (MCP)	2	
Structurarea cerințelor software pentru AI. Folosirea tooluri AI pentru code review. Folosirea tooluri AI pentru depanarea problemelor. Automatizarea sarcinilor ce țin de programare prin generarea de scripturi. Analiza codului cu tooluri bazate pe AI. Automatizarea unor sarcini simple cum ar fi actualizarea dependințelor. Generarea de teste automate folosind toolurile bazate pe AI	2	
Limitări și erori frecvente ale codului generat. Folosirea serverelor MCP. Declanșarea de sarcini automate care folosesc AI	2	
Etica și legalitatea utilizării AI în programare.	2	
Instrumente: Copilot, Replit, Cursor, Codeium, Claude Code, Codex.	2	
Studii de caz: implementarea unor algoritmi cu ajutorul AI.	2	
Integrarea AI în fluxul DevOps.	2	
Evaluarea și testarea codului generat.	2	
Proiect complex – design ghidat de AI.	2	
Recuperări, recapitulare.	2	

Bibliografie[12]
1. Pagina cursului: https://cv.upt.ro/course
2. Andrej Karpathy – Intro to Large Language Models & Prompting (cursuri și materiale video, 2023–2025).
3. DeepLearning.AI & OpenAI – ChatGPT Prompt Engineering for Developers, Coursera, 2023.
4. Brian Roemmele – Prompt Engineering Mastery, ediția 2023.
5. Columbia University – Prompt Engineering & Programming with OpenAI (syllabus oficial, 2024).
6. Stanford CS324 – Large Language Models (Lecture Notes), Stanford University, 2023.
7. Amershi et al. (Microsoft Research) – Guidelines for Human-AI Interaction, ACM, 2019.
8. Bommasani et al. – On the Opportunities and Risks of Foundation Models, Stanford CRFM, 2021 (pentru partea de etică și impact).
9. O'Reilly Media – Programming with Generative AI (colecție de articole, 2024).

8.2 Activităţi aplicative[13]	Număr de ore	Metode de predare

Bibliografie[14]

9. Evaluare

Tip activitate	**9.1** Criterii de evaluare[15]	**9.2** Metode de evaluare	**9.3** Pondere din nota finală
9.4 Curs	Cunoașterea noțiunilor de bază. Rezolvarea unor aplicații	Întrebări teorie + exerciții de analiză prompt/context.	100%
9.5 Activităţi aplicative	**S:**		
	L:		
	P[16]**:**		
	Pr:		
9.6 Standard minim de performanţă (se prezintă cunoștințele minim necesare pentru promovarea disciplinei și modul în care se verifică stăpânirea lor[17])			
• studentul să poată folosi un asistent AI pentru a implementa corect un algoritm simplu, cu verificarea rezultatului.			

Data completării	**Titular de curs (semnătura)**	**Titular activităţi aplicative (semnătura)**
21.09.2026	Ș.l.univ.dr.ing. Cătălin IAPĂ	Ș.l.univ.dr.ing. Cătălin IAPĂ
Director de departament (semnătura)	**Data avizării în Consiliul Facultăţii**[18]	**Decan (semnătura)**
Prof.univ.dr.habil.ing. Mihai MICEA		ș.l.univ.dr.ing. Robert KRISTOF