FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	Universitatea Politehnica Timișoara
1.2 Facultatea [1] / Departamentul[2]	Facultatea de Inginerie din Hunedoara / Inginerie Electrică și Informatică Industrială
1.3 Domeniul de studii (denumire/cod[3])	INGINERIE ELECTRICĂ / 10
1.4 Ciclul de studii	Master
1.5 Programul de studii (denumire/cod/calificarea)	TEHNICI INFORMATICE ÎN INGINERIA ELECTRICĂ

2. Date despre disciplină

2.1a Denumirea disciplinei/Categoria formativă[4]			Programarea avansată a interfețelor grafice utilizator/ DF				
2.1b Denumirea disciplinei în limba engleză			Advanced programming of graphical user interfaces				
2.2 Titularul activităţilor de curs			Prof. dr. ing. Pănoiu Manuela				
2.3 Titularul activităţilor aplicative[5]			Șef lucr. dr. ing. Ghiormez Loredana				
2.4 Anul de studiu[6]	I	**2.5** Semestrul	II	**2.6** Tipul de evaluare	E	**2.7** Regimul disciplinei[7]	DOB

3. Timp total estimat - ore pe semestru (activități directe (asistate integral), activități asistate parțial și activități neasistate[8])

3.1 Număr de ore asistate integral/săptămână	3.5 , din care:	ore curs	1.5	ore seminar/laborator/proiect			2
3.1* Număr total de ore asistate integral/sem.	49 , din care:	ore curs	21	ore seminar/laborator/proiect			28
3.2 Număr total de ore desfășurate on-line asistate integral/sem.	10 , din care:	ore curs	10	ore seminar/laborator/proiect			0
3.3 Număr de ore asistate parțial/saptămână	, din care:	ore proiect, cercetare		ore practică		ore elaborare lucrare de disertație	
3.3* Număr total de ore asistate parțial/ semestru	, din care:	ore proiect cercetare		ore practică		ore elaborare lucrare de disertație	
3.4 Număr de ore activități neasistate/ săptămână	5.43 , din care:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					2
		ore studiu individual după manual, suport de curs, bibliografie și notițe					1.43
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri					2
3.4* Număr total de ore activități neasistate/ semestru	76 , din care:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					28
		ore studiu individual după manual, suport de curs, bibliografie și notițe					20
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri					28
3.5 Total ore/săptămână[9]	8.93						
3.5* Total ore/semestru	125						
3.6 Număr de credite	5						

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	• Cunoștințe de bază de Programarea calculatoarelor, Programare orientată pe obiect.
4.2 de rezultate ale învățării	•

5. Condiţii (acolo unde este cazul)

5.1 de desfăşurare a cursului	• Sală de curs echipată cu videoproiector și conexiune la Internet. • Materiale suport: laptop, proiector, tablă

5.2 de desfăşurare a activităţilor practice	• Sală echipată cu videoproiector și conexiune la Internet • Laborator cu 12-14 calculatoare, tablă

6. Rezultatele învățării la formarea cărora contribuie disciplina

Cunoștințe	• C3. Absolventul sumarizează și aplică concepte privind prelucrarea datelor, rețele neuronale și algoritmi de machine learning în ingineria electrică. • C4. Absolventul descrie, compară și utilizează metodologii și instrumente informatice moderne pentru dezvoltarea și testarea aplicațiilor tehnico-industriale.
Abilități	• A1. Proiectează, implementează și testează sisteme inteligente integrate hardware–software, utilizând limbaje și medii avansate de programare. • A3. Utilizează sisteme de achiziție de date, senzori și microcontrolere în aplicații distribuite sau în timp real.
Responsabilitate și autonomie	• RA1. Gestionează activități inginerești complexe, cu asumarea responsabilității pentru deciziile luate în contexte tehnologice și organizaționale. • RA2. Se implică activ în proiecte tehnice sau de cercetare, demonstrând autonomie și capacitate de adaptare la schimbări. • RA4. Susține dezvoltarea profesională continuă prin autoevaluare și învățare independentă, adaptată nevoilor tehnologice curente.

7. Obiectivele disciplinei (asociate rezultatelor învățării specifice acumulate)

• Implementare de aplicaţii cu interfaţă grafică avansată utilizând Java și Java FX
Cunoașterea conceptelor specifice programării GUI – MVC (Model – View – Controller. Abilități de programare dirijată de evenimente în Java.
Abilități de programare interfețe grafice avansate în Java.
• Abilități de programare aplicații în rețea în Java
• Abilitati de programare utilizând Java Fx

8. Conţinuturi

8.1 Curs	Număr de ore	Din care on-line	Metode de predare
1. Introducere în programare GUI. 1.1 Scurt istoric al GUI. 1.2 Coordonatele sistem. 1.3 Paradigmele WIMP (Windows, Icons, Menus, Pointer)	2		Expunerea informațiilor esențiale pe videoproiector sau pe ZOOM Detalierea informațiilor expuse Demonstrarea utilizând un mediu de programare adecvat Conversaţia
2.Concepte, practici și criterii GUI. 2.1. Definiții și caracteristici. 2.2. Containere și componente. 2.3. Administratori de dispunere GUI și managementul administratorilor 2.4. Utilizarea componentelor SWING într-o interfața grafică 2.5 Proiectarea GUI Framework	2	2	
3. Programarea dirijată de evenimente. 3.1 Evenimente. Tipuri de evenimente 3.2 Event listners. 3.3 Concurență.	2	2	

3.4 Evenimente definite de utilizatori. 3.5 Tratarea evenimentelor în Java.			
4. Implementarea interfețelor grafice utilizator. 4.1 GUI design paterns. 4.2 Model – View – Controller 4.3 Componente Swing GUI și Graphics 2D 4.4 Proiectarea și personalizarea componentelor GUI	2		
5 Java FX 5.1. Noțiuni introductive despre JavaFX 5.2. Structura de bază a JavaFX 5.3. Panouri, controale de interfață și forme 5.4. Clase ajutatoare în JavaFx 5.5. Layout Panes	8	4	
6 JavaFx în Netbeans 6.1 Crearea unei aplicații simple JavaFX 6.2. Utilizarea CSS 6.3. Crearea unei aplicații JavaFX FXML 6.4. Construirea de interfețe grafice cu ajutorul controalelor JavaFX	2	2	
7. Construirea de interfețe grafice cu ajutorul controalelor JavaFX	3		

	Bibliografie[10] 1. Panoiu Manuela, Programarea avansată a interfețelor grafice utilizator (curs în format electronic, pe campusul virtual al UPT, https://cv.upt.ro/course/view.php?id=1286) 2. Iordan A. E., Pănoiu M., Programarea Avansata A Intefetelor Grafice Utilizator, Editura Politehnica Timișoara, 2013 3. JavaFX 4. JavaFX Documentation 5 Learn JavaFX 8: Building User Experience and Interfaces with Java 8 1st ed. Edition, Kindle Edition 6 JavaFX Tutorial

8.2 Activităţi aplicative[11]	Număr de ore	Din care on-line	Metode de predare
1. Mediul de programare NetBeans. Clase și compunerea obiectelor în limbajul Java 2. Clase și moștenire în limbajul Java	2 2		Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare în NetBeans
3.Aplicații folosind compunerea de obiecte, conceptul de moștenire, clasă care conține mai multe metode pentru interacționarea între diferite clase și testarea acestora.	2		
4. Aplicații GUI folosind pachetul de clase java.awt.	2		
5. Aplicații GUI folosind pachetul de clase javax.swing	4		
6. Programare generică în limbajul Java	4		
7. Prelucrarea fișierelor text și a celor binare în Java	4		
8. Aplicații Java care utilizează șablonul arhitectural Model – View- Controller	4		
9. Aplicații realizate în Netbeans folosind JavaFx	2		
10. Test de laborator	2		

	Bibliografie[12] 1.Ghiormez Loredana, Programarea avansată a interfețelor grafice utilizator (laborator în format electronic, pe campusul virtual al UPT , https://cv.upt.ro/course/view.php?id=1286 2. Iordan A. E., Pănoiu M., Programarea Avansata A Intefetelor Grafice Utilizator, Editura Politehnica Timișoara, 2013 3.Tanasa S., Olaru C., Java de la 0 la expert, editura Polirom, Colectia Calculatoare. Informatica, 2011. 4. Danciu. D, Mardale G., Arta programarii in Java , Editura Albastra, 2003.

9. Evaluare

Tip activitate	**9.1** Criterii de evaluare[13]	**9.2** Metode de evaluare	**9.3** Pondere din nota finală
9.4 Curs	Cunoștințe practice de implementare aplicații GUI	Examen pe campusul virtual al UPT cu acces la documentație	60%
9.5 Activităţi aplicative	**S:**		
	L: Abilităţi practice de implementare aplicații avansate GUI	Verificări periodice pe calculator, lucrare practică la final de semestru	40%
	P:		
	Pr:		
	Tc-R[14]:		

9.6 Standard minim de performanţă (volumul de cunoştinţe minim necesar pentru promovarea disciplinei şi modul în care se verifică stăpânirea lui)[15]

- *Activități aplicative:*
- Implementarea aplicațiilor pe calculator în cadrul verificărilor periodice și a lucrării practice la final de semestru în proporție de 50 % (dezvoltarea unei aplicații în Netbeans cu interfață grafică, care să realizeze minim trei prelucrări de date și afișări pe interfața grafică utilizator)
- *Curs:*
- Implementarea unei aplicații cu interfață grafică în proporție de 50% în cadrul examenului final

Data completării

10.09.2026

Titular de curs (semnătura)

Titular activităţi aplicative (semnătura)

Director de departament (semnătura)

Data avizării în Consiliul Facultăţii[16]

23.09.2026

Decan (semnătura)