Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti

Facultatea de Electronică, Telecomunicaţii şi Tehnologia Informaţiei

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	**Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti**
1.2 Facultatea	**Electronică, Telecomunicaţii şi Tehnologia Informaţiei**
1.3 Departamentul	**Electronică Aplicată şi Ingineria Informaţiei**
1.4 Domeniul de studii	Calculatoare şi Tehnologia Informaţiei
1.5 Ciclul de studii	Licenţă
1.6 Specializarea	Ingineria Informaţiei

2. Date despre disciplină

2.1 Denumirea disciplinei (ro) (en)				Programarea calculatoarelor şi limbaje de programare 3 - Proiect Python			
2.2 Titularul activităţilor de curs				Prof. Dr. Bogdan Emanuel IONESCU			
2.3 Titularul activităţilor de seminar / laborator				Prof. Dr. Bogdan Emanuel IONESCU			
2.4 Anul de studiu	2	2.5 Semestrul	I	2.6. Tipul de evaluare	V	2.7 Regimul disciplinei	Op
2.8 Tipul disciplinei		F	2.9 Codul disciplinei	04.F.03.A.009		2.10 Tipul de notare	Nota

3. Timpul total estimat (ore pe semestru al activităţilor didactice)

3.1 Număr de ore pe săptămână	1	Din care: 3.2 curs	0.00	3.3 seminar/laborator	1
3.4 Total ore din planul de învăţământ	14.00	Din care: 3.5 curs	0	3.6 seminar/laborator	14
Distribuţia fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie şi notiţe Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate Pregătire seminarii/ laboratoare/proiecte, teme, referate, portofolii şi eseuri					9
Tutorat					0
Examinări					2
Alte activităţi (dacă există):					0
3.7 Total ore studiu individual	11.00				
3.8 Total ore pe semestru	25				
3.9 Numărul de credite	1				

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	• Programarea calculatoarelor şi limbaje de programare 1 • Programarea calculatoarelor şi limbaje de programare 2
4.2 de rezultate ale învăţării	Cunoştinţe generale de programarea calculatoarelor.

5. Condiţii necesare pentru desfăşurarea optimă a activităţilor didactice (acolo unde este cazul)

5.1 Curs	

5.2 Seminar/ Laborator/Proiect	Sistem de videoproiecţie pentru prezenţa fizică şi acces la platforma Microsoft Teams pentru consultaţii online. Prezenţa la şedinţele de proiect este obligatorie, conform regulament ETTI.

6. Obiectiv general *(Se referă la intenţiile profesorilor pentru studenţi, la ceea ce studenţii vor fi învăţaţi în timpul cursului. Oferă o orientare cu privire la locul cursului în cadrul domeniului ştiinţific abordat, precum şi la rolul pe care acesta îl are în cadrul specializării studiate. Vor fi descrise de o manieră generală tematicile abordate, justificarea includerii cursului în planul de învăţământ al specializării studiate etc.)*

- Formarea competenţelor de utilizare a limbajului Python în dezvoltarea de aplicaţii.
- Înţelegerea conceptelor fundamentale de programare şi aplicarea lor în proiecte reale.
- Dezvoltarea capacităţii de analiză, proiectare şi implementare a soluţiilor software.
- Exersarea lucrului în echipă şi a prezentării proiectelor software.

7. Competenţe *(Capacitatea dovedită de a utiliza cunoştinţe, aptitudini şi abilităţi personale, sociale şi/sau metodologice în situaţii de muncă sau de studiu şi pentru dezvoltarea profesională şi personală. Reflectă cerinţele angajatorilor.)*

Specifice	• Aplicarea cunoştinţelor, conceptelor şi metodelor de bază privitoare la arhitecturasistemelor de calcul, microprocesoare, microcontrollere, limbaje şi tehnici deprogramare
Transversale (generale)	• Analiza metodică a problemelor întâlnite în activitate, identificând elementelepentru care există soluţii consacrate, asigurând astfel îndeplinirea sarcinilorprofesionale. • Adaptarea la noile tehnologii, dezvoltarea profesională şi personală, prin formarecontinuă folosind surse de documentare tipărite, software specializat şi resurseelectronice în limba română şi, cel puţin, într-o limbă de circulaţie internaţională.

8. Rezultatele învăţării *(Sunt enunţuri sintetice referitoare la ceea ce un student va fi capabil să facă sau să demonstreze la finalizarea unui curs. Rezultatele învăţării reflectă realizările studentului şi mai puţin intenţiile profesorului. Rezultatele învăţării informează studenţii despre ceea ce se aşteaptă de la ei din punct de vedere al performanţei, pentru a obţine notele şi creditele dorite. Sunt definite în termeni concreţi, folosind verbe similare exemplelor de mai jos şi indică ceea ce se va urmări prin evaluare. Rezultatele învăţării vor fi astfel redactate încât să fie evidenţiată clar relaţia faţă de competenţele definite la punctul 7.)*

Cunoştinţe	*Rezultatul asimilării de informaţii prin învăţare. Cunoştinţele reprezintă ansamblul de fapte, principii, teorii şi practici legate de un anumit domeniu de muncă sau de studiu. Pot fi teoretice şi/sau faptice.* • însuşirea programării în limbajul de programare Python • însuşirea programării unei interfeţe grafice în Python • însuşirea cunoştinţelor de a integra baze de date şi API-uri externe într-o aplicaţie Python • însuşirea cunoştinţelor de a depana un program în limbajul Python.

Aptitudini	*Capacitatea de a aplica cunoştinţe şi de a utiliza know-how pentru a duce la îndeplinire sarcini şi a rezolva probleme. Aptitudinile sunt descrise ca fiind cognitive (implicând utilizarea gândirii logice, intuitive şi creative) sau practice (implicând dexteritate manuală şi utilizarea de metode, materiale, unelte şi instrumente).* • abilitatea de a înţelege şi explica un program scris în cod Python, • abilitatea de a utiliza API-uri externe în Python, • abilitatea de a identifica soluţii de programare, • abilitatea de a înţelege fluxul operaţional şi de a depana programe în Python, • abilitatea de a comunica şi argumenta soluţii.
Responsabilitate şi autonomie	*Capacitatea cursantului de a aplica în mod autonom şi responsabil cunoştinţele şi aptitudinile sale.* • Capacitatea de a selecta şi parcurge surse bibliografice, • Capacitatea de a promova şi contribui cu soluţii noi, • Capacitatea de a învăţa concepte noi, • Capacitatea de comunicare a informaţiilor cu alţi colegi, • Dezvoltarea autonomiei în procesul de învăţare.

9. Metode de predare *(Se vor avea în vedere metode care să asigure predarea centrată pe student. Se va descrie modul în care se asigură participarea studenţilor la stabilirea propriului parcurs de învăţare, cum se identifică eventualele rămâneri în urmă şi ce măsuri remediale se adoptă în astfel de cazuri.)*

Proiectul se bazează pe programarea în echipă a unei aplicaţii în limbajul Python. Fiecare echipă alege o temă, alocă fiecărui membru o sarcină specifică, dezvoltă o interfaţă grafică şi integrează toate submodulele dezvoltate într-o aplicaţie finală, utilizând concepte de programare orientată pe obiecte, baze de date şi utilizare de API-uri externe.

10. Conţinuturi

PROIECT		
Nr. crt.	**Conţinutul**	**Nr. ore**
1	Introducere în Python. Tipuri de date, operatori, structuri de control, funcţii simple. Exemple: mini-calculator, conversii unităţi.	2
2	Definirea temelor de proiect. Alegerea temei şi stabilirea cerinţelor. Exemple: chatbot, aplicaţie buget, organizator reţete, joc de memorie.	2
3	Structura aplicaţiei şi backend. Implementarea funcţionalităţilor de bază, testare iniţială în linie de comandă.	2
4	Interfaţă şi OOP. Crearea interfeţei grafice, conectarea cu backend, introducere în OOP (clase, moştenire).	2
5	Funcţionalităţi avansate şi baze de date. Persistenţa datelor, conectarea la baze de date, integrarea de API-uri externe.	2
6	Optimizare şi testare. Debugging, optimizarea codului, pregătirea proiectului final.	2
7	Evaluare finală. Prezentarea proiectelor şi discuţii asupra soluţiilor implementate.	2
	Total:	14
Bibliografie:		

11. Evaluare

Tip activitate	11.1 Criterii de evaluare	11.2 Metode de evaluare	11.3 Pondere din nota finală
11.4 Curs			
11.5 Seminar/laborator/proiect	Abilitatea de a modela şi rezolva programatic cu succes o problemă în Python.	Evaluarea se realizează pe baza activitatăţii pe parcursul semestrului şi a prezenţei la laborator.	50
	Verificarea finală a proiectului şi a documentaţiei.	Evaluarea finală a proiectului Python.	50
11.6 Condiţii de promovare			
Obţinerea a 50% din punctajul total.			

12. Coroborarea conţinutului disciplinei cu aşteptările reprezentanţilor angajatorilor şi asociaţiilor profesionale reprezentative din domeniul aferent programului, precum şi cu stadiul actual al cunoaşterii în domeniul ştiinţific abordat şi practicile în instituţii de învăţământ superior din Spaţiul European al Învăţământului Superior (SEÎS)

În ultimii ani Python a devenit cel mai popular limbaj de programare, reprezentând pilonul dezvoltării inteligenţei artificiale. Pe lângă aceasta, există numeroase alte aplicaţii legate de statistică, big data, procesări de baze de date sau dezvoltare de aplicaţii pentru social-media în care Python reprezintă limbajul de programare cel mai des utilizat. Acest proiect vine în completarea pregătirii studenţilor electronişti pentru câmpul muncii. Bazându-se pe cunoştinţele acumulate în urma acestei discipline, studenţii pot realiza o multitudine de sarcini în acest limbaj de programare, de la specificarea cerinţelor la execuţia, depanarea şi interpretarea rezultatelor.

Data completării	Titular de curs	Titular(i) de aplicaţii
20.10.2025		Prof. Dr. Bogdan Emanuel Ionescu

Data avizării în departament	Director de departament

Data aprobării în Consiliul Facultăţii	Decan