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	**Telecomunicaţii**
1.4 Domeniul de studii	Inginerie Electronică, Telecomunicaţii şi Tehnologii Informaţionale
1.5 Ciclul de studii	Licenţă
1.6 Specializarea	Reţele şi Software de Telecomunicaţii

2. Date despre disciplină

2.1 Denumirea disciplinei (ro) (en)				Tehnologii de programare în Internet			
2.2 Titularul activităţilor de curs				S.l./Lect. Dr. Laurentiu BOICESCU			
2.3 Titularul activităţilor de seminar / laborator				S.l./Lect. Dr. Laurentiu BOICESCU			
2.4 Anul de studiu	3	2.5 Semestrul	2	2.6. Tipul de evaluare	E	2.7 Regimul disciplinei	Ob
2.8 Tipul disciplinei		D	2.9 Codul disciplinei	04.D.06.O.312		2.10 Tipul de notare	Nota

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

3.1 Număr de ore pe săptămână	3	Din care: 3.2 curs	2	3.3 seminar/laborator	1
3.4 Total ore din planul de învăţământ	42	Din care: 3.5 curs	28	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					4
Tutorat					0
Examinări					4
Alte activităţi (dacă există):					0

3.7 Total ore studiu individual	8.00
3.8 Total ore pe semestru	50
3.9 Numărul de credite	2

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	Parcurgerea şi/sau promovarea următoarelor discipline: Programarea Calculatoarelor Structuri de Date si Algoritmi Programare Obiect-Orientată

4.2 de rezultate ale învăţării	Acumularea următoarelor cunoştinţe: Cunoştinţe generale de programare Lucrul cu structuri de date, referinţe, obiecte, clase Scrierea programelor obiect-orientate

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

5.1 Curs	Conform regulamentului studiilor universitare în UPB. Cursul se va desfăşura într-o sală dotată cu videoproiector şi computer.
5.2 Seminar/ Laborator/Proiect	Laboratorul se va desfăşura într-o sală cu dotare specifică, care trebuie să includă: videoproiector, sisteme de calcul cu mediu de dezvoltare NetBeans IDE, Server web (Apache Tomcat), Server de baze de date (MySQL), hipervizor pentru masini virtuale (VirtualBox)

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.)*

Disciplina are ca obiectiv familiarizarea studenţilor cu tehnologii de programare larg utilizate în Internet: desktop (comunicaţii la nivel socket TCP şi UDP), Web (tehnologii client şi server HTTP), mobile (Android, Windows Phone, etc.), acces la baze de date şi servicii web, folosind diferite limbaje de programare şi reprezentare: HTML, CSS, JavaScript, Java, PHP, C#, XML, SQL.
Se urmăresc: completarea pregătirii studenţilor în domeniul ingineriei software, cu accent pe tehnologiile de programare în Internet, dobândirea cunoştinţelor necesare proiectării şi implementării de aplicaţii software pentru Internet, realizarea de aplicaţii practice, utilizând noţiunile studiate în cadrul cursului.

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	C3. Aplicarea cunoştinţelor, conceptelor şi metodelor de bază privitoare la arhitectura sistemelor de calcul, microcontrolere, limbaje şi tehnici de programare C3.4 Elaborarea de programe într-un limbaj de programare obiect-orientata, pornind de la specificarea cerinţelor si pana la execuţia, depanarea si interpretarea rezultatelor în corelaţie cu procesorul utilizat C4. Elaborarea specificaţiilor tehnice, achiziţia, instalarea şi exploatarea echipamentelor de comunicaţii, fixe şi mobile, precum şi planificarea, configurarea şi integrarea serviciilor de telecomunicaţii şi elemente de securitatea informaţiei - abilităţile de aplicare a cunoştinţelor generale privind tehnologiile software avansate (Java, Qt/C++, .NET/C#, PHP, MySQL, XML, etc.) pentru realizarea serviciilor şi sistemelor (Web, mobile, etc.) de comunicaţie C6. Utilizarea limbajelor de programare şi instrumentelor specializate pentru inginerie software, cu orientate către sistemele de comunicaţii integrate C6.1 Definirea unor metodologii, limbaje şi instrumente software implicate în dezvoltarea sistematică a sistemelor software de comunicaţii C6.4 Utilizarea tehnicilor orientate pe obiecte pentru analiza si modelarea sistemelor SW C6.5 Programarea unor elemente pentru aplicaţii funcţionând in reţea şi WEB

Transversale (generale)	CT1. Analiza metodică a problemelor întâlnite în activitate, identificând elementele pentru care există soluţii consacrate, asigurând astfel îndeplinirea sarcinilor profesionale CT3. Adaptarea la noile tehnologii, dezvoltarea profesională şi personală, prin formare continuă folosind surse de documentare tipărite, software specializat şi resurse electronice 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.* **principalele limbaje de programare şi marcare utilizate în dezvoltarea platforme web (Java, HTML, CSS,** Javascript, SQL) **modele de programare bazate pe comunicaţii stratificate şi stiva TCP/IP** **metodologii de inginerie software (proiectare, implementare şi depanare a platformelor web)** **modele arhitecturale utilizate la dezvoltarea platformelor web**
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).* Utilizează argumentat principii specifice în vederea dezvoltării aplicaţiilor web. Lucrează în echipă. Elaborează un text ştiinţific. Verifică experimental soluţii identificate. Rezolvă aplicaţii practice. Analizează şi compară tehnologii şi biblioteci pentru aplicaţii web. Identifică soluţii şi elaborează planuri de rezolvare/proiecte. Argumentează soluţiile identificate/modurile de rezolvare.
Responsabilitate şi autonomie	*Capacitatea cursantului de a aplica în mod autonom şi responsabil cunoştinţele şi aptitudinile sale.* Respectă principiile de etică academică, citând corect sursele bibliografice utilizate. Demonstrează receptivitate pentru contexte noi de învăţare. Manifestă colaborare cu ceilalţi colegi şi cadre didactice în desfăşurarea activităţilor didactice Demonstrează autonomie în organizarea situaţiei/contextului de învăţare sau a situaţiei problemă de rezolvat Conştientizează valoarea contribuţiei sale în domeniul ingineriei la identificarea de soluţii viabile/sustenabile care să rezolve probleme din viaţa socială şi economică (responsabilitate socială). Analizează şi valorifică oportunităţi de afaceri/de dezvoltare antreprenorială în domeniul de specialitate. Demonstrează abilităţi de management al situaţiilor din viaţa reală (gestionarea timpului, colaborare, conflict).

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 studenților la stabilirea propriului parcurs de învățare, cum se identifică eventualele rămâneri în urmă și ce măsuri remediale se adoptă în astfel de cazuri.)*

Pornindu-se de la analiza caracteristicilor de învăţare ale studenţilor şi de la nevoile lor specifice, procesul de predare va explora metode de predare atât expozitive (prelegerea, expunerea), cât şi conversative-interactive, bazate pe modele de învăţare prin descoperire facilitate de explorarea directa şi indirectă a realităţii (experimentul, demonstraţia, modelarea), dar şi pe metode bazate pe acţiune, precum exerciţiul, activităţile practice şi rezolvarea de probleme.
În activitatea de predare vor fi utilizate prelegeri, în baza unor prezentări Power Point care vor fi puse la dispoziţia studenţilor. Fiecare curs va debuta cu recapitularea capitolelor deja parcurse, cu accent asupra noţiunilor parcurse la ultimul curs. Primele cursuri vor recapitula noţiunile de programare orientată pe obiecte studiate în semestrele precedente.
Prezentările utilizează imagini şi scheme, astfel încât informaţiile prezentate să fie uşor de înţeles şi asimilat.
Acestă disciplină oferă informaţii şi activităţi practice menite să-i sprijine pe studenţi în eforturile de învăţare şi de dezvoltare a unor relaţii optime de colaborare şi comunicare într-un climat favorabil învăţării prin descoperire.
Se va încuraja abilitatea de lucru în echipă pentru rezolvarea diferitelor sarcini de învăţare.

10. Conţinuturi

CURS		
Capitolul	**Conţinutul**	**Nr. ore**
1	Introducere în tehnologiile de programare în Internet -Introducere în reţelele de comunicaţii -Modele de comunicaţii stratificate; protocoale de comunicaţie; stiva TCP/IP -Introducere în Internet şi tehnologii utilizate: DNS, HTTP	4
2	Introducere in tehnologiile desktop Java. -Introducere în limbajul de programare Java; convenţii de programare de bază -Cuvinte cheie în limbajul Java, tipuri de date, structuri de control -Programare cu orientare pe obiecte în Java, clase, polimorfism -Fluxuri de intrare şi ieşire -Programarea la nivel socket	6
3	Tehnologii de programare Web de tip server -Modelul client-server pentru aplicaţii web -Protocolul HTTP -Tehnologia Java Servlet -Asigurarea persistenţei în cadrul aplicaţiilor web. Sesiuni şi tehnologia cookie -Tehnologia JSP (Java ServerPages) -Programare web avansată. Delegare şi redirecţionare a cererilor -Modelele arhitecturale de proiectare MV (Model-View) si MVC (Model-View-Controller) -Limbajul SQL. Baze de date. Aplicaţii practice -Integrarea bazelor de date în aplicaţiile web	12
4	Tehnologii de programare Web de tip client -Introducere în tehnologiile web de tip client -Limbajele HTML şi CSS. Realizarea interfeţelor grafice pentru aplicaţii web. -Limbajul Javascript. Interfeţe web dinamice.	6

	Total:	28
Bibliografie:		

LABORATOR		
Nr. crt.	**Conţinutul**	**Nr. ore**
1	Familiarizarea cu mediile de dezvoltare -Famiiliarizare cu NetBeans IDE 6.1 -Recapitulare programare orientată pe obiecte Java	2
2	Programarea aplicatiilor Java la nivel socket -Fire de execuţie -Socket-uri in Java	2
3	Programarea serviciilor Web folosind Java Servlet -Tehnologia Java Servlet -Servlet-uri HTTP	2
4	Programarea aplicaţiilor Web folosind Java ServerPages -Tehnologia JSP -Modelul arhitectural MV (Model-View) -Modelul arhitectural MVC (Model-View-Controller)	2
5	Stocarea datelor aplicaţiilor web în baze de date MySQL -Limbajul SQL. Sintaxa MySQL -Utilizarea MySQL in cadrul aplicatiilor Java	2
6	Tehnologii de programare Web de tip client -Limbajele HTML, CSS şi JavaScript -Integrarea interfeţelor HTML în aplicaţii Java Web	2
7	Colocviu final de laborator	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	- cunoaşterea noţiunilor teoretice fundamentale; - cunoaşterea modului de aplicare a teoriei la probleme specifice; - analiza tehnicilor şi metodelor teoretice.	Verificare în timpul semestrului la date fixate la începutul semestrului. Verificare finala în timpul sesiunii.	40%

11.5 Seminar/laborator/proiect	- cunoaşterea modului de concepere ale unor programe şi dezvoltarea aptitudinilor de proiectare a acestora - demonstrarea funcţionarii unui program implementat	Colocviu final de laborator, cuprinzând o componentă teoretică şi o componentă practică, prin verificarea modului de rezolvare (implementare, testare, funcţionare) de către student a unei probleme practice.	30%
	Teme de casa: - capacitatea de analiză şi sinteză a temei abordate - capacitatea de implementare a aplicaţiei practice, conform cerinţelor - capacitatea de prezentare şi argumentare soluţiei practice implementate	Susţinerea finală a proiectului realizat	30%
11.6 Condiţii de promovare			
Obţinerea a 50% din punctajul total. Obţinerea a 50% din punctajul aferent activităţii pe parcursul semestrului.			

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)

Prin activităţile desfăşurate, studenţii dobândesc abilitatea de a oferi soluţii unor probleme şi de a propune idei de îmbunătăţire a situaţiei existente în domeniul ingineriei software, mai exact în ramura dezvoltării aplicaţiilor pentru Internet.
În dezvoltarea conţinutului disciplinei s-au avut în vedere cunoştinţe, aspecte teoretice şi practice, precum şi fenomene descrise de literatura de specialitate, cercetările proprii publicate şi experienţa titularilor disciplinei.
Prin activitatea de disciplină se are în vedere dezvoltarea abilităţilor absolventului de a proiecta şi implementa noi aplicaţii software pentru Internet, precum şi de a analiza şi depana aplicaţii software existente, inclusiv în vederea îmbunătăţirii acestora. În acest fel, absolvenţii cursului pot contribui la îmbunătăţirea mediului economic, în domeniul aplicaţiilor pentru Internet.

Data completării	Titular de curs	Titular(i) de aplicaţii
21.09.2026	S.l./Lect. Dr. Laurentiu BOICESCU	S.l./Lect. Dr. Laurentiu BOICESCU

Universitatea Națională de Știință și Tehnologie Politehnica București

Facultatea de Electronică, Telecomunicații și

Tehnologia Informației

Data avizării în departament	Director de departament
22.09.2026	Conf. Dr. Serban Georgica Obreja
Data aprobării în Consiliul Facultății	Decan
22.09.2026	Prof. Dr. Mihnea Udrea