



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei	REȚELE DE CALCULATOARE – TIDS301						
1.2 Titularul activităților de curs	Lect. Dr. Damian Felicia Anișoara						
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator	Lect. Dr. Damian Felicia Anișoara						
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	E	1.8 Regimul disciplinei	Ob DS

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	4	din care: 2.2 curs	2	2.3 seminar/laborator	0/2
2.4 Total ore din planul de învățământ	56	din care: 2.5 curs	28	2.6 seminar/laborator	0/28
Distribuția fondului de timp					ore
2.7 Activități monitorizate					-
Practică					-
Elaborare proiect diplomă					-
2.8 Activități individuale					19
Documentare					5
Studiu individual					5
Referate					5
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					4
2.9 Total ore pe semestru					75
2.10 Numărul de credite					3

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Arhitectura Sistemelor de Calcul (TIDS214) — organizarea procesorului, interfețe hardware-software; Teoria Transmisiunii Informației (TIDS212) — codificare, modulații, canal de comunicație; Sisteme de Operare (TIDS302) — gestiunea proceselor, I/O, interfața cu rețeaua.
3.2 de competențe	Cunoașterea modelelor OSI și TCP/IP la nivel conceptual; înțelegerea principiilor de adresare și rutare IP; familiarizarea cu linia de comandă Linux/Windows pentru configurarea rețelei (ifconfig/ipconfig, ping, traceroute); abilitatea de a interpreta capturi de pachete simple.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu tablă, videoproiector și acces la internet; platforma Sakai pentru distribuirea materialelor; recomandare: instalarea Cisco Packet Tracer și Wireshark înainte de curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator de informatică cu calculatoare individuale; Cisco Packet Tracer instalat; Wireshark instalat; studenții pot folosi propriul laptop; platforma Sakai pentru predarea lucrărilor.

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C11. Deține cunoștințe specializate privind arhitecturile rețelelor de calculatoare locale și extinse, protocoalele de comunicații (TCP/IP, OSI) și tehnologiile de transmisiune a datelor, cu conștientizarea evoluțiilor recente în domeniu.
5.2 Aptitudini	A11. Proiectează, configurează și administrează rețele de calculatoare locale și extinse (LAN/WAN), implementând protocoale de comunicații, topologii de rețea și soluții de management al traficului de date.
5.3 Responsabilitate și autonomie	R11. Își asumă responsabilitatea pentru migrarea și integrarea sistemelor informatice complexe în infrastructuri cloud sau distribuite, gestionând riscurile tehnice asociate și asigurând continuitatea serviciilor pe parcursul tranziției.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Introducere și modele de referință. Evoluția rețelelor: de la ARPANET la Internet. Clasificarea: LAN, MAN, WAN, PAN. Topologii: magistrală, stea, inel, plasă. Modelul OSI — 7 niveluri, funcții, interfețe. Modelul TCP/IP — 4 niveluri, comparație cu OSI. Standardizare: ISO, IEEE, IETF, RFC.	2	Prelegere interactivă	
Tema 2: Nivelul fizic. Suporturi de transmisie: cablu torsadat (UTP Cat5e/Cat6/Cat6a), fibră optică (single-mode, multi-mode), wireless. Codificarea datelor: NRZ, Manchester, 4B/5B. Multiplexare: TDM, FDM, WDM. Modulații digitale: ASK, FSK, PSK, QAM. Teoremele Nyquist și Shannon: debitul maxim, capacitatea canalului.	2	Prelegere interactivă	
Tema 3: Nivelul legătură de date. Delimitarea cadrelor, controlul erorilor (paritate, CRC), controlul fluxului. Coduri Hamming. Protocoale cu fereastră glisantă: Go-Back-N, Selective Repeat. Accesul multiplu: ALOHA, CSMA/CD, CSMA/CA.	2	Prelegere interactivă	
Tema 4: Ethernet și rețele LAN. IEEE 802.3: Ethernet 10/100/1000 Mbps, 10 Gigabit. Structura cadrului Ethernet. Switch-ul: tabela MAC, learning, flooding. VLAN — IEEE 802.1Q: configurare, trunk. Spanning Tree Protocol (STP). Comparație Ethernet vs. Wi-Fi.	2	Prelegere interactivă	
Tema 5: Rețele wireless. IEEE 802.11 (Wi-Fi): 802.11a/b/g/n/ac/ax (Wi-Fi 6). Arhitectura WLAN: BSS, ESS, AP. CSMA/CA, RTS/CTS. Securitate: WEP, WPA2, WPA3. Bluetooth și BLE. Rețele mobile: 4G LTE, 5G NR. IoT și LPWAN: LoRa, Zigbee, NB-IoT.	2	Prelegere interactivă	
Tema 6: Adresarea IP. Adresarea IPv4: structura, clase, subnetting, CIDR. Calculul subrețelelor: mască, adresă rețea, broadcast,	2	Prelegere, calcule numerice	

hosturi. NAT: tipuri și funcționare. Adresarea IPv6: structura (128 biți), tipuri (unicast, multicast, anycast). Coexistența IPv4/IPv6: dual stack, tunneling, NAT64.			
Tema 7: Rutarea. Algoritmi: distanță-vector (Bellman-Ford), stare legătură (Dijkstra). Protocoale: RIP, OSPF, BGP. Rutare statică vs. dinamică. Sisteme autonome (AS), IGP vs. EGP. ECMP. SDN: separarea planului de control de cel de date.	2	Prelegere interactivă	
Tema 8: Nivelul transport. UDP: structura, utilizări. TCP: conexiunea (three-way handshake), terminarea. Controlul fluxului: fereastra glisantă. Controlul congestiei: slow start, congestion avoidance, fast retransmit/recovery (Reno, CUBIC, BBR). Multiplexarea porturilor.	2	Prelegere interactivă	
Tema 9: Protocoale de suport. DHCP: procesul DORA, relay. DNS: ierarhie, tipuri de înregistrări (A, AAAA, MX, CNAME, NS, PTR), rezoluție recursivă/iterativă. ARP și RARP. ICMP: ping, traceroute. NDP în IPv6.	2	Prelegere interactivă	
Tema 10: Nivelul aplicație. HTTP/1.1, HTTP/2, HTTP/3 (QUIC). HTTPS și TLS 1.3. FTP, SFTP, SSH. Email: SMTP, POP3, IMAP. DNS over HTTPS/TLS. WebSockets și gRPC. REST vs. GraphQL.	2	Prelegere interactivă	
Tema 11: Securitatea rețelelor. Amenințări: sniffing, spoofing, MITM, DDoS, ARP poisoning. Firewall: filtrare pachete, stateful, next-generation. IDS/IPS. VPN: IPsec, SSL VPN, WireGuard. Criptografia în rețele: TLS, certificate X.509, PKI. Zero Trust.	2	Prelegere interactivă	
Tema 12: Rețele cloud și virtualizare. VLAN, VXLAN, GRE tunneling. SDN și OpenFlow. NFV. Rețele cloud: AWS VPC, Azure VNet, Google Cloud VPC. Kubernetes networking: pods, services, ingress, CNI plugins.	2	Prelegere interactivă	
Tema 13: Monitorizarea rețelelor. SNMP v3: MIB, operații, securitate. NetFlow/sFlow/IPFIX. Syslog, NTP, LLDP. Instrumente: Wireshark, tcpdump, Nagios, Zabbix, Grafana + Prometheus. Troubleshooting OSI layer-by-layer.	2	Prelegere interactivă	
Tema 14: Tendințe și recapitulare. SD-WAN, Intent-Based Networking. 5G și edge computing: latență ultra-scăzută, network slicing. Rețele cuantice: principii și stadiul cercetării. Recapitulare generală.	2	Prelegere, recapitulare	
	Bibliografie obligatorie 1. Tanenbaum, A.S., Feamster, N., Wetherall, D.J., Computer Networks, 6th ed., Pearson, 2021. 2. Kurose, J.F., Ross, K.W., Computer Networking: A Top-Down Approach, 8th ed., Pearson, 2021. 3. Forouzan, B.A., Data Communications and Networking, 5th ed., McGraw-Hill, 2013. Bibliografie complementară 4. Stevens, W.R., TCP/IP Illustrated, Vol. 1: The Protocols, 2nd ed., Addison-Wesley, 2011.		

	<p>5. Comer, D.E., Internetworking with TCP/IP, Vol. 1, 6th ed., Pearson, 2013.</p> <p>Resurse software:</p> <p>6. Cisco Packet Tracer — simulator de rețele (cont gratuit NetAcad): https://www.netacad.com/courses/packet-tracer</p> <p>7. Wireshark — analizor de protocol, documentație: https://www.wireshark.org/docs/</p> <p>8. RFC Editor — standardele protocoalelor: https://www.rfc-editor.org/</p>		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Packet Tracer — introducere. Instalarea și configurarea Cisco Packet Tracer. Construirea unei rețele simple: 3 PC-uri conectate printr-un switch. Configurarea adreselor IP. Testarea cu ping. Analiza traficului ARP și ICMP în modul simulare.	2	Lucrare practică Packet Tracer	
Lucrarea L2: Subnetting IPv4. Calculul manual al subnetting-ului: mască, adresă rețea, broadcast, intervale de hosturi. Exerciții VLSM: dimensionarea subrețelelor pentru cerințe date. Configurarea subrețelelor în Packet Tracer și verificarea conectivității.	2	Lucrare practică, calcul	
Lucrarea L3: VLAN. Configurarea a 3 VLAN-uri pe un switch Cisco: porturi de acces și trunk. Verificarea izolării traficului. Vizualizarea tabelii MAC și a porturilor trunk.	2	Lucrare practică Packet Tracer	
Lucrarea L4: Inter-VLAN routing. Configurarea inter-VLAN routing cu router-on-a-stick (subinterfețe 802.1Q). Verificarea rutării între VLAN-uri. Configurarea DHCP per VLAN.	2	Lucrare practică Packet Tracer	
Lucrarea L5: Rutare statică. Configurarea rutării statice între 3 routere. Configurarea rutei default. Simularea căderii unui link și actualizarea manuală a tabelii de rutare.	2	Lucrare practică Packet Tracer	
Lucrarea L6: Rutare dinamică RIP și OSPF. Configurarea RIP v2 pe o topologie cu 3 routere. Configurarea OSPF single-area pe aceeași topologie. Compararea timpilor de convergență și a metricilor.	2	Lucrare practică Packet Tracer	
Lucrarea L7: DHCP și DNS. Configurarea unui server DHCP în Packet Tracer: pool, gateway, DNS. Verificarea atribuirii automate. Captură Wireshark a procesului DORA. Analiza unui query DNS și identificarea tipurilor de înregistrări.	2	Lucrare practică Wireshark	
Lucrarea L8: Analiza traficului TCP. Captură Wireshark a unui three-way handshake TCP: identificarea SYN, SYN-ACK, ACK și a numerelor de secvență. Urmărirea ferestrei de congestione pe un transfer de fișier.	2	Lucrare practică Wireshark	
Lucrarea L9: HTTP și HTTPS. Captură Wireshark a unui schimb HTTP: request, response, headere. Compararea cu HTTPS: handshake TLS și date criptate. Analiza certificatului digital.	2	Lucrare practică Wireshark	
Lucrarea L10: Adresare IPv6. Configurarea adreselor IPv6 statice și prin SLAAC în Packet Tracer. Configurarea rutării OSPFv3. Ping IPv6. Configurarea dual stack (IPv4 + IPv6).	2	Lucrare practică Packet Tracer	

Lucrarea L11: Securitate — ACL. Configurarea unui ACL standard și a unuia extins pe un router Cisco: filtrarea traficului. Testarea regulilor. Configurarea port security pe switch.	2	Lucrare practică Packet Tracer	
Lucrarea L12: Securitate — SSH și VPN. Configurarea SSH pe routere Cisco: cheie RSA, utilizatori locali, eliminarea Telnet. Configurarea unui VPN site-to-site IPsec simplu între două locații.	2	Lucrare practică Packet Tracer	
Lucrarea L13: Monitorizarea rețelei. Capturi Wireshark simultane: ICMP, ARP, DNS — analiză comparativă. Filtre BPF și display filters avansate. Utilizarea tcpdump în linie de comandă.	2	Lucrare practică Wireshark, Linux	
Lucrarea L14: Proiect integrat și colocviu. Proiectarea și implementarea în Packet Tracer a unei rețele de firmă: 3 departamente (VLAN-uri), server DHCP, rutare OSPF, ACL de securitate și acces Internet prin NAT. Documentarea topologiei. Colocviu oral.	2	Proiect individual, colocviu	
	<p>Bibliografie obligatorie laborator</p> <ol style="list-style-type: none"> 1. Tanenbaum, A.S., Feamster, N., Wetherall, D.J., Computer Networks, 6th ed., Pearson, 2021 — exercițiile practice. 2. Cisco Packet Tracer — documentație: https://www.netacad.com/courses/packet-tracer 3. Wireshark — ghid utilizare și filtre: https://www.wireshark.org/docs/wsug_html_chunked/ <p>Bibliografie complementara laborator</p> <ol style="list-style-type: none"> 4. Kurose, J.F., Ross, K.W., Computer Networking: A Top-Down Approach, 8th ed., Pearson, 2021 — laboratoarele interactive Wireshark. 5. Stevens, W.R., TCP/IP Illustrated, Vol. 1: The Protocols, 2nd ed., Addison-Wesley, 2011. <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. Cisco Packet Tracer — simulator gratuit: https://www.netacad.com/courses/packet-tracer 7. Wireshark — analizor de protocol: https://www.wireshark.org/ 8. Cisco Learning Network — exerciții de rețele: https://learningnetwork.cisco.com/ 9. Subnet Calculator — calculator online subnetting: https://www.subnet-calculator.com/ 		

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

Rețelele de Calculatoare furnizează competențele tehnice esențiale oricărui inginer TI care lucrează cu infrastructura modernă: configurarea și depanarea rețelelor LAN/WAN, înțelegerea stivei TCP/IP la nivel de protocol, securizarea rețelelor și migrarea serviciilor în cloud. Aceste competențe sunt solicitate direct în rolurile de administrator de rețea, inginer DevOps, arhitect cloud și specialist în securitate cibernetică.

Componenta de laborator, bazată pe Cisco Packet Tracer și Wireshark, urmărește formarea competenței practice de proiectare, configurare și depanare a rețelelor pe topologii reale — direct corelată cu A11 din matricea de corelare. Proiectul integrat din L14 reproduce un scenariu industrial complet, iar responsabilitatea pentru integrarea în infrastructuri cloud (R11) este acoperită în Temele 12 și 14.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea și aplicarea modelelor OSI și TCP/IP; calculul corect al subnetting-ului IPv4/IPv6 (VLSM); cunoașterea funcționării protocoalelor Ethernet, ARP, IP, TCP, UDP, DHCP, DNS, HTTP/S; cunoașterea algoritmilor de rutare și a protocoalelor RIP, OSPF, BGP; cunoașterea principalelor amenințări și mecanisme de securitate (firewall, VPN, TLS).	Examen scris (2 ore): 1 problemă de subnetting IPv4 cu VLSM (calculul complet pentru 3 subrețele cu cerințe diferite), 1 problemă de analiză a unui schimb de protocoale (identificarea nivelurilor OSI și câmpurilor dintr-un cadru/segment dat) și 1 problemă de rutare (construirea tabelului de rutare cu algoritmul Dijkstra pentru o topologie dată)	40%
8.5 Seminar/laborator	Funcționalitatea configurațiilor Packet Tracer (conectivitate end-to-end verificată, protocoale configurate corect); calitatea analizei Wireshark (identificarea corectă a câmpurilor, interpretarea secvenței de protocoale); calitatea proiectului integrat din L14 (topologie completă, documentație, funcționalitate demonstrată); calitatea prezentării la colocviu.	Evaluare continuă: configurații Packet Tracer și rapoarte Wireshark predate pe Sakai după fiecare lucrare (50%) și proiect integrat cu colocviu oral individual la L14 (50%)	60%
8.6 Standard minim de performanță:			
Rezolvarea corectă și completă la examenul scris a problemei de subnetting IPv4 (calculul tuturor câmpurilor pentru toate subrețelele cerute, cu justificarea alegerii măștilor) — aceasta verificând stăpânirea competenței fundamentale a disciplinei. La colocviul oral din L14: demonstrarea ping end-to-end în proiectul Packet Tracer între hosturi din subrețele diferite, cu explicarea rutei parcurse de pachet la nivel de adrese MAC și IP prin fiecare dispozitiv intermediar.			

Data completării 17.03.2026	Semnătura titularului de curs Lect. Dr. Damian Felicia Anișoara	Semnătura titularului de seminar / laborator Lect. Dr. Damian Felicia Anișoara
Data avizării în Consiliul Departamentului 24.03.2026	Semnătura directorului DIT Lect. dr. Antohe Valerian	
Data aprobării în Consiliul Facultății 07.04.2026	Semnătura decanului FMDT Conf. dr. Beteringhe Adrian	



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei	TEORIA SISTEMELOR – TIDS302						
1.2 Titularul activităților de curs	Lect. Dr. Panfiloiu Gheorghe						
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator	Lect. Dr. Panfiloiu Gheorghe						
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	E	1.8 Regimul disciplinei	Ob DS

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	3	din care: 2.2 curs	2	2.3 seminar/laborator	0/1
2.4 Total ore din planul de învățământ	42	din care: 2.5 curs	28	2.6 seminar/laborator	0/14
Distribuția fondului de timp					ore
2.7 Activități monitorizate					-
Practică					-
Elaborare proiect diplomă					-
2.8 Activități individuale					19
Documentare					5
Studiu individual					6
Referate					4
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					4
2.9 Total ore pe semestru					75
2.10 Numărul de credite					3

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Matematici Discrete (TIDS205) — algebră liniară, transformata Laplace, ecuații diferențiale; Metode Numerice (TIDS211) — rezolvarea numerică a ecuațiilor diferențiale; Teoria Transmisiunii Informației (TIDS212) — transformata Fourier, analiza spectrală; Electronică Digitală (TIDS209) — circuite analogice și sisteme în buclă.
3.2 de competențe	Cunoașterea transformatei Laplace și a proprietăților sale; familiarizarea cu ecuațiile diferențiale liniare de ordinul 1 și 2; cunoașterea transformatei Fourier și a analizei spectrale; abilitatea de a utiliza MATLAB pentru calcule numerice de bază.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu tablă pentru demonstrații matematice, videoproiector și acces la internet; platforma Sakai; recomandare: MATLAB sau Octave instalat, cu Control System Toolbox, înainte de curs.
-------------------------------	--

4.2 de desfășurare a seminarului/laboratorului	Laborator de informatică cu calculatoare individuale; MATLAB cu Control System Toolbox sau Octave gratuit cu pachetul control; studenții pot folosi propriul laptop; acces la MATLAB Online ca alternativă.
--	---

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C29. Deține cunoștințe specializate privind teoria sistemelor — sisteme liniare, funcții de transfer, stabilitate, răspuns în frecvență — și aplicarea acestora în modelarea și analiza sistemelor de control și a microsistemelor.
5.2 Aptitudini	A29. Modelează și analizează sisteme dinamice utilizând tehnici din teoria sistemelor (funcții de transfer, diagrame Bode, spații de stare), utilizând instrumente software specializate (MATLAB/Simulink) pentru simularea comportamentului sistemelor de control.
5.3 Responsabilitate și autonomie	R29. Își asumă responsabilitatea pentru modelarea corectă a sistemelor dinamice în proiectele ingineresti, validând experimental modelele teoretice și documentând transparent discrepanțele dintre comportamentul simulat și cel real.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Introducere în teoria sistemelor. Sistemul ca noțiune fundamentală: intrare, stare, ieșire, mediu. Clasificarea sistemelor: liniare/nelineare, invariante/variante în timp, continue/discrete, SISO/MIMO. Modelarea matematică: ecuații diferențiale, ecuații de stare. Transformata Laplace: definiție, proprietăți, transformate uzuale, teoremele valorii inițiale și finale.	2	Prelegere interactivă	
Tema 2: Funcția de transfer. Definiția funcției de transfer: raportul transformatei Laplace a ieșirii față de intrare, în condiții inițiale nule. Poli și zerouri: semnificație fizică, reprezentare grafică în planul complex. Funcții de transfer de ordinul 1 și 2: parametrii caracteristici. Răspunsul impuls (funcția Green) și răspunsul treaptă. Legătura funcție de transfer — ecuație diferențială.	2	Prelegere, demonstrații matematice	
Tema 3: Sisteme de ordinul 1 și 2. Sistemul de ordinul 1: constanta de timp T , răspunsul la treaptă, răspunsul la rampă. Sistemul de ordinul 2: pulsația proprie ω_0 , factorul de amortizare ζ . Cazuri: supramorforit ($\zeta > 1$), critic ($\zeta = 1$), submorforit ($\zeta < 1$). Indicatorii de calitate ai răspunsului tranzitoriu: timpul de creștere, de stabilire, suprareglajul. Sisteme de ordin superior: dominanța polilor.	2	Prelegere, calcule numerice	
Tema 4: Diagrame de flux de semnal și schema bloc. Schema bloc: elemente (bloc, sumator, ramificație), simplificarea schemelor bloc (serie, paralel, buclă). Diagrama de flux de semnal (signal flow graph): noduri, ramuri, câștiguri de ramură. Formula lui Mason pentru câștigul total. Aplicații: modelarea sistemelor electronice și mecanice.	2	Prelegere interactivă	

<p>Tema 5: Analiza sistemelor în buclă închisă. Structura standard a sistemului de reglare automată: regulatorul, procesul, senzorul, perturbația. Funcția de transfer în buclă închisă: formula generală. Efectul perturbațiilor și al zgomotului de măsurare. Eroarea în regim permanent pentru intrări tip treaptă, rampă, parabolă: tipul sistemului (tip 0, 1, 2) și constantele de eroare.</p>	2	Prelegere interactivă	
<p>Tema 6: Stabilitatea sistemelor liniare. Definiția stabilității: BIBO (Bounded Input Bounded Output) și stabilitatea Lyapunov. Condiția necesară și suficientă: toți polii în semiplana stângă ($\text{Re}(p) < 0$). Criteriul Routh-Hurwitz: matricea Routh, cazurile speciale (rândul de zerouri, rândul complet nul). Aplicații: determinarea domeniului parametrilor de stabilitate.</p>	2	Prelegere, demonstrații matematice	
<p>Tema 7: Locul rădăcinilor. Principiul locului rădăcinilor (Evans): variația polilor în buclă închisă în funcție de un câștig K. Regulile de trasare: punctele de plecare/sosire, asimptotele, unghiurile de plecare/sosire, intersecția cu axa imaginară. Interpretarea locului rădăcinilor: stabilitate, tip de răspuns, alegerea câștigului. Locul rădăcinilor cu MATLAB: rlocus.</p>	2	Prelegere interactivă, MATLAB	
<p>Tema 8: Răspunsul în frecvență — diagrama Bode. Funcția de răspuns în frecvență: $H(j\omega)$. Diagrama Bode: modulul în dB și faza în grade vs. frecvența în decibeli pe decadă. Diagramele Bode asimptotice: contribuția fiecărui tip de factor (pol real, zero real, pol complex, zero complex). Câștigul static. Verificarea cu MATLAB: bode.</p>	2	Prelegere interactivă, MATLAB	
<p>Tema 9: Diagrame Nyquist și Nichols. Criteriul de stabilitate Nyquist: diagrama polară $H(j\omega)$, conturul Nyquist, numărul de înconjurări al punctului (-1, 0). Marginea de câștig și marginea de fază din diagrama Nyquist. Diagrama Nichols: modulul vs. faza, grila constantelor funcției în buclă închisă. Compararea criteriilor de stabilitate în frecvență.</p>	2	Prelegere interactivă	
<p>Tema 10: Reprezentarea în spațiul stărilor. Ecuațiile de stare: $\dot{x} = Ax + Bu$, $y = Cx + Du$. Matricele sistemului: A, B, C, D — semnificație fizică. Trecerea funcție de transfer — spațiu de stare și invers (forma canonică de comandă, observare). Soluția ecuațiilor de stare: matricea de tranziție de stare e^{At}. Valorile proprii ale lui A și stabilitatea.</p>	2	Prelegere, demonstrații matematice	
<p>Tema 11: Controlabilitate și observabilitate. Controlabilitatea: definiție, matricea de controlabilitate Qc, condiția de rang.</p>	2	Prelegere interactivă	

Observabilitatea: definiție, matricea de observabilitate Q_0 , condiția de rang. Teorema Kalman de decompunere canonică. Semnificația practică: un sistem necontrolabil nu poate fi stabilizat prin feedback de stare; un sistem neobservabil nu poate fi estimat.			
Tema 12: Proiectarea reguletoarelor PID. Regulatorul PID: structura (proporțional P, integral I, derivativ D), funcția de transfer. Efectul fiecărei componente asupra răspunsului: P — câștig, I — eroare nulă în regim permanent, D — anticipare. Metode de acordare: Ziegler-Nichols (în buclă deschisă și în buclă închisă). Limitele practice ale derivatorului: filtrul derivativ.	2	Prelegere interactivă	
Tema 13: Sisteme discrete. Eșantionarea: teorema Shannon-Nyquist, frecvența de eșantionare. Transformata Z: definiție, proprietăți, transformate uzuale. Funcția de transfer discretă $H(z)$. Stabilitatea sistemelor discrete: polii în interiorul cercului unitar. Discretizarea unui sistem continuu: metoda Euler, metoda Tustin (bilinear). Regulatorul PID discret.	2	Prelegere interactivă	
Tema 14: Simulare în MATLAB/Simulink — recapitulare. Utilizarea Control System Toolbox: tf, ss, bode, nyquist, rlocus, step, impulse. Crearea unui model Simulink pentru un sistem de control cu PID. Simularea răspunsului și compararea cu calculul analitic. Validarea modelului: discrepanțe simulate vs. real — documentarea lor. Recapitulare generală.	2	Prelegere, MATLAB/Simulink	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> Ogata, K., Modern Control Engineering, 5th ed., Pearson, 2010. Voicu, M., Teoria sistemelor, Ed. Academiei Române, București, 2008. Mînză, V. et al., Théorie des systèmes — éléments fondamentaux, Fondation Académica, 2003. <p>Bibliografie complementară</p> <ol style="list-style-type: none"> Franklin, G.F., Powell, J.D., Emami-Naeini, A., Feedback Control of Dynamic Systems, 8th ed., Pearson, 2019. Nise, N.S., Control Systems Engineering, 8th ed., Wiley, 2019. <p>Resurse software:</p> <ol style="list-style-type: none"> MathWorks Control System Toolbox — documentație și tutoriale: https://www.mathworks.com/help/control/ GNU Octave — alternativă gratuită la MATLAB, pachetul control: https://octave.org/ MATLAB Online — acces MATLAB în browser (cont MathWorks): https://matlab.mathworks.com/ 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Transformata Laplace și funcția de transfer în MATLAB. Reprezentarea funcțiilor de transfer cu tf(). Calculul și afișarea polilor și zerourilor (pzmap). Trasarea răspunsului la treaptă (step) și la impuls (impulse) pentru sisteme de ordinul 1 și 2 cu parametri variabili. Verificarea analitică a parametrilor: T, ω_0 , ζ .	2	Lucrare practică MATLAB	
Lucrarea L2: Simplificarea schemelor bloc și formula Mason. Implementarea în MATLAB	2	Lucrare practică MATLAB	

a sistemelor descrise prin scheme bloc (series, parallel, feedback). Calculul funcției de transfer a unui sistem cu 3 blocuri, sumator și buclă negativă — compararea cu rezultatul formulei Mason calculat manual. Trasarea răspunsului la treaptă al sistemului compus.			
Lucrarea L3: Stabilitate — Routh-Hurwitz și locul rădăcinilor. Verificarea stabilității cu criteriul Routh-Hurwitz pentru 3 sisteme de ordinul 3 și 4 (calculul manual al tabloului Routh). Trasarea locului rădăcinilor cu rlocus() pentru un sistem de ordinul 3. Alegerea câștigului K optim pentru un suprareglaj dat. Verificarea cu step().	2	Lucrare practică MATLAB	
Lucrarea L4: Diagrame Bode și marjele de stabilitate. Trasarea diagramelor Bode cu bode() pentru 3 sisteme diferite. Citirea manuală a marginii de câștig (Gm) și a marginii de fază (Pm) de pe diagramă. Verificarea cu margin(). Efectul unui zero suplimentar asupra diagramei Bode: compararea cu sistemul fără zero.	2	Lucrare practică MATLAB	
Lucrarea L5: Spațiul stărilor. Conversia funcție de transfer — spațiu de stare cu tf2ss() și ss2tf(). Calculul matricei de tranziție de stare cu expm(A*t). Verificarea controlabilității și observabilității cu ctrb() și obsv() — calculul rangului. Simularea răspunsului cu initial() (condiții inițiale nenule).	2	Lucrare practică MATLAB	
Lucrarea L6: Proiectarea regulatorului PID. Acordarea manuală a unui PID cu metoda Ziegler-Nichols (în buclă deschisă). Implementarea PID cu pid() sau tf(). Compararea răspunsurilor P, PI, PID pentru un proces dat. Optimizarea automată cu pidtune(). Evaluarea performanțelor: suprareglaj, timp de stabilire, eroare în regim permanent.	2	Lucrare practică MATLAB	
Lucrarea L7: Sisteme discrete și proiect integrat. Discretizarea unui sistem continuu cu c2d() (metoda Tustin, T_e = 0.1s). Compararea răspunsului discret cu cel continuu. Trasarea diagramei poli-zerouri în planul Z. Proiect integrat: modelarea completă a unui sistem de control de ordinul 2 cu PID — funcție de transfer, loc al rădăcinilor, Bode, răspuns la treaptă, spațiu de stare. Colocviu oral.	2	Lucrare practică MATLAB, colocviu	
	Bibliografie obligatorie laborator 1. Ogata, K., Modern Control Engineering, 5th ed., Pearson, 2010 — exercițiile MATLAB. 2. MathWorks Control System Toolbox — documentație completă: https://www.mathworks.com/help/control/ 3. MathWorks Getting Started with Control System Toolbox: https://www.mathworks.com/help/control/getting-started-with-control-system-toolbox.html		

	<p>Bibliografie complementara laborator</p> <p>4. Franklin, G.F., Powell, J.D., Emami-Naeini, A., Feedback Control of Dynamic Systems, 8th ed., Pearson, 2019 — laboratoarele MATLAB.</p> <p>5. GNU Octave Documentation — pachetul control: https://octave.sourceforge.io/control/</p> <p>Resurse online gratuite recomandate</p> <p>6. MATLAB Online — acces gratuit cu cont MathWorks: https://matlab.mathworks.com/</p> <p>7. GNU Octave — alternativă gratuită la MATLAB: https://octave.org/</p> <p>8. MathWorks Control System Toolbox Tutorials — tutoriale video și exemple: https://www.mathworks.com/help/control/examples.html</p> <p>9. MIT OpenCourseWare 6.302 — Feedback Systems, note de curs: https://ocw.mit.edu/courses/6-302-feedback-systems-spring-2007/</p>
--	---

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

Teoria Sistemelor furnizează cadrul matematic unificator pentru înțelegerea comportamentului dinamic al oricărui sistem fizic: electric, mecanic, termic, hidraulic sau software. Competențele formate — analiza stabilității, proiectarea PID, modelarea în spațiul stărilor — sunt esențiale în robotică, sisteme embedded cu buclă de control, sisteme de achiziție și automatizare industrială. Componenta de laborator, realizată integral în MATLAB/Control System Toolbox, urmărește formarea capacității de a simula, analiza și proiecta sisteme de control real, de a valida modelele teoretice prin simulare și de a documenta transparent discrepanțele — direct corelată cu A29 și R29 din matricea de corelare.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea și aplicarea funcțiilor de transfer pentru sisteme de ordinul 1 și 2; aplicarea criteriului Routh-Hurwitz pentru analiza stabilității; trasarea și interpretarea locului rădăcinilor; trasarea și interpretarea diagramei Bode cu calculul marginilor de câștig și de fază; cunoașterea reprezentării în spațiul stărilor și a condițiilor de controlabilitate/observabilitate; proiectarea regulatorului PID cu metoda Ziegler-Nichols.	Examen scris (2 ore): 1 problemă de calcul al funcției de transfer în buclă închisă cu calculul erorilor în regim permanent, 1 problemă de analiză a stabilității cu criteriul Routh-Hurwitz (tabloul complet cu concluzia) și 1 problemă de trasare asimptotică a diagramei Bode cu citirea marginilor de stabilitate	40%
8.5 Seminar/laborator	Corectitudinea codului MATLAB și a rezultatelor grafice din lucrările de laborator	Evaluare continuă: cod MATLAB și grafice predate pe Sakai după fiecare lucrare (50%)	60%

	(grafice etichetate, parametri citați corect din grafice, verificarea analitică); calitatea proiectului integrat din L7 (toate reprezentările prezente: funcție de transfer, pzmap, bode, step, rlocus, spațiu de stare); calitatea prezentării și a justificărilor la colocviu.	și proiect integrat cu colocviu oral individual la L7 (50%)	
--	--	---	--

8.6 Standard minim de performanță:

Rezolvarea corectă și completă la examenul scris a problemei de stabilitate Routh-Hurwitz (tabloul complet, numărul corect de rădăcini cu parte reală pozitivă și concluzia de stabilitate) — aceasta verificând stăpânirea celui mai direct instrument de analiză a stabilității din disciplină. La colocviul oral din L7: citirea corectă a marginii de câștig și a marginii de fază din diagrama Bode obținută în MATLAB, cu explicarea semnificației fizice a fiecărei marje și a consecințelor atunci când acestea sunt negative.

Data completării	Semnătura titularului de curs Lect. Dr. Panfiloiu Gheorghe	Semnătura titularului de seminar / laborator Lect. Dr. Panfiloiu Gheorghe
17.03.2026		
Data avizării în Consiliul Departamentului	Semnătura directorului DIT Lect. dr. Antohe Valerian	
24.03.2026		
Data aprobării în Consiliul Facultății	Semnătura decanului FMDT Conf. dr. Beteringhe Adrian	
07.04.2026		



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei	SISTEME DE OPERARE – TIDS303						
1.2 Titularul activităților de curs	Lect. Dr. Burlacu Cătălina Mercedes						
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator	Lect. Dr. Burlacu Cătălina Mercedes						
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	E	1.8 Regimul disciplinei	Ob DS

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	4	din care: 2.2 curs	3	2.3 seminar/laborator	0/1
2.4 Total ore din planul de învățământ	56	din care: 2.5 curs	42	2.6 seminar/laborator	0/14
Distribuția fondului de timp					ore
2.7 Activități monitorizate					-
Practică					-
Elaborare proiect diplomă					-
2.8 Activități individuale					44
Documentare					10
Studiu individual					15
Referate					15
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					4
2.9 Total ore pe semestru					100
2.10 Numărul de credite					4

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Arhitectura Sistemelor de Calcul (TIDS214) — organizarea procesorului, ierarhia memoriei, interfețe hardware-software; Programare Orientată pe Obiecte (TIDS204) — limbajul C/C++, pointeri, alocare dinamică; Rețele de Calculatoare (TIDS301) — interfața SO cu rețeaua, socket-uri.
3.2 de competențe	Cunoașterea limbajului C la nivel mediu: pointeri, alocare dinamică (malloc/free), fișiere; familiarizarea cu linia de comandă Linux de bază (navigarea sistemului de fișiere, procesele, redirectările); înțelegerea conceptului de proces și a diferenței față de program.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu tablă pentru demonstrații, videoproiector și acces la internet; platforma Sakai; recomandare: o mașină virtuală Linux (Ubuntu 22.04 LTS sau echivalent) configurată înainte de curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator de informatică cu calculatoare individuale; mașini virtuale Linux instalate (VirtualBox/VMware cu Ubuntu 22.04 LTS);

	compilatorul GCC, utilitarele strace, ltrace, valgrind, perf; studenții pot folosi propriul laptop cu VM sau WSL2 pe Windows.
--	---

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C6. Posedă cunoștințe critice privind sistemele de operare moderne — gestiunea proceselor, memoriei, fișierelor și securității — și capacitatea de a evalua comparativ soluții alternative.
5.2 Aptitudini	A10. Administrează și configurează sisteme de operare în medii complexe, gestionând procesele, resursele de memorie, sistemele de fișiere și politicile de securitate ale sistemului.
5.3 Responsabilitate și autonomie	R6. Contribuie activ la revizuirea performanței strategice a echipei tehnice, identificând punctele slabe ale procesului de dezvoltare și propunând măsuri concrete de îmbunătățire a productivității și calității livrabilelor.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Introducere în sistemele de operare. Definiția și rolul SO: interfața dintre hardware și software. Evoluția SO: mainframe, time-sharing, PC, mobil, cloud. Structura internă a SO: monolitic (Linux), microkernel (MINIX, QNX), hibrid (Windows NT), exokernel. Apeluri de sistem (syscall): mecanismul trap, user mode vs. kernel mode. Comparatie Linux, Windows, macOS, Android.	3	Prelegere interactivă	
Tema 2: Gestiunea proceselor. Procesul: definiție, spațiu de adrese, stările procesului (running, ready, blocked, zombie). PCB (Process Control Block): câmpurile esențiale. Operații pe procese în Linux: fork(), exec(), wait(), exit(). Zombi și orfan: cauze și consecințe. Context switch: costul hardware și software. Thread-uri vs. procese: avantaje și dezavantaje. POSIX threads (pthreads): pthread_create, pthread_join.	3	Prelegere interactivă	
Tema 3: Planificarea proceselor. Criterii de planificare: utilizarea CPU, throughput, turnaround time, waiting time, response time. Algoritmi nepreemptivi: FCFS, SJF. Algoritmi preemptivi: Round Robin, SRTF. Algoritmi cu priorități: priorități statice și dinamice, starvation și aging. Planificarea multi-level queue și multi-level feedback queue. CFS (Completely Fair Scheduler) din Linux: virtual runtime, red-black tree.	3	Prelegere interactivă	
Tema 4: Sincronizarea proceselor. Problema secțiunii critice: condițiile de corectitudine (excludere mutuală, progres, așteptare limitată). Soluții hardware: instrucțiunile atomice test-and-set, compare-and-swap, fetch-and-add. Mutex (lock): implementarea cu spinlock, sleep lock. Semafoare: binar și de numărare, implementare, utilizare. Variabile	3	Prelegere interactivă	

de condiție: wait, signal, broadcast. Deadlock: definiție, condițiile necesare (Coffman).			
Tema 5: Deadlock — detecție, prevenire, evitare. Graful de alocare a resurselor: procesul, resursa, ciclul. Strategii de tratare: ignorare (politica struțului), prevenire (negarea unei condiții Coffman), evitare (algoritmul bancherului Dijkstra), detecție și recuperare. Algoritmul bancherului: starea sigură, cererea de resurse, exemplu numeric. Recuperarea din deadlock: terminarea proceselor, preempțarea resurselor.	3	Prelegere, demonstrații numerice	
Tema 6: Gestiunea memoriei — fundamente. Spațiul de adrese al procesului: cod, date, heap, stac. Relocarea: statică vs. dinamică. Registrele base și limit. Swapping: principii și limitări. Fragmentarea internă și externă. Compactarea memoriei. Alocarea memoriei contigue: first fit, best fit, worst fit. Buddy system: principiu și implementare.	3	Prelegere interactivă	
Tema 7: Memoria virtuală — paginare. Paginarea: pagini, cadre, tabela de pagini, traducerea adreselor virtuale în fizice. TLB (Translation Lookaside Buffer): funcționare, hit rate, costul miss-ului. Tabele de pagini ierarhice: 2 și 4 niveluri (x86-64). Tabele de pagini inversate. Page fault: cauze, tratarea în kernel. Demand paging: principiu, performanță.	3	Prelegere interactivă	
Tema 8: Algoritmi de înlocuire a paginilor. Principiul localității: spațiale și temporale. Thrashing: cauze și remedii, working set. Algoritmi: optimal (OPT/Belady), FIFO (și anomalia Belady), LRU (implementare exactă vs. aproximări), Clock (NRU), LFU. Clock de două mâini din Linux. Compararea algoritmilor pe un exemplu numeric. Segmentarea: avantaje față de paginare, segmentare paginată.	3	Prelegere, calcule numerice	
Tema 9: Sisteme de fișiere. Conceptele de bază: fișier, director, cale. Operații pe fișiere: open, read, write, seek, close. Structuri de date: FCB (File Control Block), tabela de fișiere deschise. Organizarea directoarelor: single-level, two-level, arborescent, cu link-uri. Implementarea sistemului de fișiere: alocare contiguă, cu liste înlănțuite, indexată (inode). FAT (File Allocation Table): structura, avantaje, limitări.	3	Prelegere interactivă	
Tema 10: Sisteme de fișiere Linux și Windows. Ext4: structura (superblock, group descriptor, inode table, data blocks), jurnalizare (journaling), extente. NTFS: MFT (Master File Table), jurnalizare, compresie, criptare EFS. ZFS: copy-on-write, RAID-Z, snapshot-uri. Montarea sistemelor de fișiere:	3	Prelegere interactivă	

VFS (Virtual File System) în Linux. Comparație ext4 vs. NTFS vs. ZFS: performanță, fiabilitate, funcționalități.			
Tema 11: Gestiunea I/O. Ierarhia I/O: dispozitiv fizic, controller, driver, SO, aplicație. Tehnici de transfer: polling, întreruperi, DMA. Buffering și caching la nivel SO. Spooling. Driver-ul de dispozitiv: structura, interfața cu kernel-ul. Planificarea acceselor pe disc: FCFS, SSTF, SCAN, C-SCAN, C-LOOK. SSD vs. HDD: implicații pentru planificarea I/O.	3	Prelegere interactivă	
Tema 12: Securitatea sistemelor de operare. Modelul de protecție: domenii de protecție, matricea de acces, capability lists, ACL (Access Control Lists). Controlul accesului în Linux: utilizatori, grupuri, permisiuni rwx, setuid/setgid, sticky bit. SELinux și AppArmor: MAC (Mandatory Access Control) vs. DAC (Discretionary Access Control). Amenințări: overflow de buffer, privilege escalation, race condition. Mecanisme de apărare: ASLR, stack canary, DEP/NX.	3	Prelegere interactivă	
Tema 13: Virtualizarea și containerizarea. Mașini virtuale: tipul 1 (hypervisor bare-metal: VMware ESXi, KVM) și tipul 2 (hosted: VirtualBox, VMware Workstation). Tehnici de virtualizare: trap-and-emulate, binary translation, paravirtualizare, virtualizare hardware (Intel VT-x, AMD-V). Containere Docker: namespace-uri Linux (PID, NET, MNT, UTS, IPC, USER), cgroups. Comparație VM vs. container: izolare, overhead, portabilitate.	3	Prelegere interactivă	
Tema 14: SO în medii distribuite și recapitulare. Sistemele de operare distribuite: transparența locației, migrarea proceselor. NFS (Network File System) și SMB/CIFS: montarea la distanță. Sincronizarea distribuită: ceasuri logice Lamport, algoritmul lui Ricart-Agrawala. Sisteme de operare mobile: Android (Linux kernel, Binder IPC, ART runtime), iOS (XNU kernel). Recapitulare generală.	3	Prelegere, recapitulare	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> 1. Arpaci-Dusseau, R.H., Arpaci-Dusseau, A.C., Operating Systems: Three Easy Pieces, v. 1.10, Arpaci-Dusseau Books, 2023. Disponibil gratuit la: https://ostep.org/ 2. Silberschatz, A., Galvin, P.B., Gagne, G., Operating System Concepts, 10th ed., Wiley, 2018. 3. Tanenbaum, A.S., Bos, H., Modern Operating Systems, 4th ed., Pearson, 2014. <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Love, R., Linux Kernel Development, 3rd ed., Addison-Wesley, 2010. 5. Kerrisk, M., The Linux Programming Interface, No Starch Press, 2010. <p>Resurse software:</p> <ol style="list-style-type: none"> 6. OSTEP — carte gratuită online și proiecte de laborator: https://ostep.org/ 7. Linux man pages — referință completă a apelurilor de sistem: https://man7.org/linux/man-pages/ 		

6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Procese și apeluri de sistem. Scrierea unui program C care utilizează fork(), exec() și wait(). Observarea PID-urilor cu ps și pstree. Utilizarea strace pentru capturarea apelurilor de sistem ale unui proces. Analiza descriptorilor de fișiere moșteniți după fork(). Simularea unui proces zombie.	2	Lucrare practică Linux/C	
Lucrarea L2: Thread-uri și sincronizare. Implementarea unei probleme producer-consumer cu pthreads: un thread producător, doi consumatori, buffer circular de dimensiune N. Sincronizarea cu mutex și variabile de condiție. Demonstrarea race condition-ului fără sincronizare. Măsurarea speedup-ului față de implementarea single-thread.	2	Lucrare practică Linux/C	
Lucrarea L3: Planificarea proceselor. Simularea în Python a algoritmilor FCFS, SJF preemptiv (SRTF) și Round Robin pentru un set dat de procese (burst time, arrival time, priority). Calculul turnaround time, waiting time și response time pentru fiecare algoritm. Afișarea diagramei Gantt. Compararea algoritmilor numeric.	2	Lucrare practică Python	
Lucrarea L4: Gestiunea memoriei. Simularea în Python a algoritmilor de înlocuire a paginilor OPT, FIFO și LRU pentru un șir de referințe de pagini dat. Calculul numărului de page fault-uri pentru fiecare algoritm și pentru dimensiuni diferite ale memoriei fizice (3, 4, 5 cadre). Demonstrarea anomaliei Belady pentru FIFO.	2	Lucrare practică Python	
Lucrarea L5: Sistemul de fișiere Linux. Examinarea structurii inode cu stat și ls -li. Crearea și urmărirea hard link-urilor și a symlink-urilor. Verificarea utilizării spațiului cu df și du. Montarea și demontarea unui sistem de fișiere (image .img) cu mount. Analiza unui sistem de fișiere ext4 cu debugfs: vizualizarea inode-urilor și a blocurilor.	2	Lucrare practică Linux	
Lucrarea L6: Securitate și permisiuni. Configurarea permisiunilor ACL extinse cu setfacl și getfacl. Demonstrarea escaladei de privilegii prin setuid: un program C setuid root care execută o comandă privilegiată. Analiza cu strace a apelurilor de sistem legate de securitate: open(), stat(), access(). Utilizarea chroot pentru izolarea unui proces.	2	Lucrare practică Linux	
Lucrarea L7: Containere și proiect integrat. Lucrul cu containere Docker: pull, run, exec, ps, stop, rm, Dockerfile, build. Compararea izolării containerelor față de procese normale: namespace-uri (unshare) și cgroups (limitarea	2	Lucrare practică Docker, C, colocviu	

CPU și memoriei). Proiect integrat: implementarea unui shell minimal în C cu fork/exec/wait, pipe-uri și redirectări. Colocviu oral.			
	<p>Bibliografie obligatorie laborator</p> <ol style="list-style-type: none"> 1. Arpaci-Dusseau, R.H., Arpaci-Dusseau, A.C., Operating Systems: Three Easy Pieces, v. 1.10, Arpaci-Dusseau Books, 2023 — proiectele practice: https://github.com/remzi-arpacidusseau/ostep-projects 2. Linux man pages — referință apeluri de sistem: https://man7.org/linux/man-pages/ 3. Kerrisk, M., The Linux Programming Interface, No Starch Press, 2010 — capitolele de procese, fișiere și sincronizare. <p>Bibliografie complementara laborator</p> <ol style="list-style-type: none"> 4. Love, R., Linux Kernel Development, 3rd ed., Addison-Wesley, 2010. 5. Docker Documentation — ghid complet: https://docs.docker.com/ <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. OSTEP — carte gratuită și proiecte: https://ostep.org/ 7. Linux man pages online: https://man7.org/linux/man-pages/ 8. Docker Documentation: https://docs.docker.com/ 9. MIT 6.S081 xv6 — sistem de operare educațional simplu: https://pdos.csail.mit.edu/6.828/2023/xv6.html 		

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

<p>Sistemele de Operare acoperă stratul software care face posibilă orice altă activitate de programare: gestionarea proceselor și a memoriei, sistemele de fișiere, securitatea și virtualizarea. Orice inginer TI — indiferent dacă scrie aplicații, administrează servere, lucrează cu containere Docker sau optimizează performanța — trebuie să înțeleagă cum SO gestionează resursele hardware pentru a scrie cod eficient și sigur.</p> <p>Componenta de laborator, realizată în Linux cu C și Python, urmărește formarea capacității de administrare și configurare a sistemelor de operare în medii complexe — direct corelată cu A10 din matricea de corelare. Proiectul integrat din L7 (shell minimal cu fork/exec/wait) reproduce mecanismul central al oricărui SO și verifică înțelegerea profundă a gestiunii proceselor.</p>

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	<p>Cunoașterea stărilor și a operațiilor pe procese (fork, exec, wait); cunoașterea algoritmilor de planificare (FCFS, SJF, RR, CFS) cu calculul metricilor (turnaround, waiting, response time); cunoașterea mecanismelor de sincronizare (mutex, semafor, variabile de condiție) și a condițiilor de deadlock; cunoașterea paginării (traducerea adreselor, TLB) și a algoritmilor de înlocuire; cunoașterea structurii sistemelor de fișiere (inode, ext4) și a</p>	<p>Examen scris (2 ore): 1 problemă de planificare a proceselor (calculul complet al tabelii Gantt și al metricilor pentru un algoritm preemptiv la alegere), 1 problemă de gestiune a memoriei (calculul page fault-urilor pentru un algoritm de înlocuire dat cu un șir de referințe) și 1 subiect de analiză comparativă (compararea a două structuri/algoritmi SO cu avantaje, dezavantaje și domenii de aplicabilitate)</p>	40%

	mecanismelor de securitate (ACL, ASLR, setuid).		
8.5 Seminar/laborator	Funcționalitatea programelor C din laborator (compilare fără erori, comportament corect verificat cu strace sau direct); corectitudinea simulărilor Python (rezultate corecte pentru toate cazurile de test); calitatea proiectului integrat din L7 (shell funcțional cu fork/exec/wait, pipe-uri și cel puțin două redirectări); calitatea prezentării și a justificărilor la colocviu.	Evaluare continuă: implementări individuale predate pe Sakai după fiecare lucrare (50%) și proiect integrat cu colocviu oral individual la L7 (50%)	60%
8.6 Standard minim de performanță:			
Rezolvarea corectă și completă la examenul scris a problemei de planificare a proceselor — diagrama Gantt completă cu toate intervalele de timp și calculul corect al celor 3 metrici pentru fiecare proces — aceasta verificând stăpânirea competenței centrale a capitolului de gestionare a proceselor. La colocviul oral din L7: demonstrarea unui shell minimal funcțional care execută cel puțin o comandă cu pipe (ex: <code>ls grep .c</code>) și explicarea pas cu pas a apelurilor <code>fork()/exec()/wait()</code> implicate.			

Data completării	Semnătura titularului de curs Lect. Dr. Burlacu Cătălina Mercedes	Semnătura titularului de seminar / laborator Lect. Dr. Burlacu Cătălina Mercedes
18.03.2026		
Data avizării în Consiliul Departamentului	Semnătura directorului DIT Lect. Dr. Antohe Valerian	
24.03.2026		
Data aprobării în Consiliul Facultății	Semnătura decanului FMDT Conf. Dr. Beteringhe Adrian	
07.04.2026		



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei	ACHIZIȚIA ȘI PRELUCRAREA DATELOR – TIDS304						
1.2 Titularul activităților de curs	Conf. Dr. Răileanu Alina-Beatrice						
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator	Lect. Dr. Anghelache Iulia						
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	E	1.8 Regimul disciplinei	Ob DS

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	3	din care: 2.2 curs	2	2.3 seminar/laborator	0/1
2.4 Total ore din planul de învățământ	42	din care: 2.5 curs	28	2.6 seminar/laborator	0/14
Distribuția fondului de timp					ore
2.7 Activități monitorizate					-
Practică					-
Elaborare proiect diplomă					-
2.8 Activități individuale					19
Documentare					5
Studiu individual					6
Referate					4
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					4
2.9 Total ore pe semestru					75
2.10 Numărul de credite					3

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Teoria Transmisiunii Informației (TIDS212) — transformata Fourier, analiza spectrală, zgomotul; Proiectarea cu Microprocesoare (TIDS215) — interfețe ADC/DAC, SPI, I2C, UART; Metode Numerice (TIDS211) — filtrare digitală, interpolare, integrare numerică; Decizie și Estimare (TIDS222) — estimarea statistică, filtrarea Kalman.
3.2 de competențe	Cunoașterea transformatei Fourier și a conceptului de spectru al unui semnal; înțelegerea principiilor conversiei analog-digitale (ADC); familiarizarea cu interfețele seriale (SPI, I2C, UART) de la disciplina Proiectarea cu Microprocesoare; abilitatea de a scrie programe Python pentru prelucrarea numerică a datelor.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu tablă pentru demonstrații matematice, videoproiector și acces la internet; platforma Sakai; recomandare: Python 3.10+ cu NumPy, SciPy, Matplotlib și nidaqmx instalate înainte de curs.
-------------------------------	---

4.2 de desfășurare a seminarului/laboratorului	Laborator dotat cu: plăci de achiziție NI USB-6001 sau echivalent (sau Arduino/STM32 cu ADC); senzori de temperatură (LM35, PT100), presiune (BMP280), accelerometru (MPU-6050); osciloscop digital; generator de funcții; calculatoare cu Python și NI-DAQmx sau LabVIEW instalate; studenții pot folosi propriul laptop.
--	--

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C30. Cunoaște principiile achiziției și prelucrării datelor — conversie analog-digitală, eșantionare, filtrare numerică, senzori și transductoare — cu conștientizarea critică a limitelor de precizie și a surselor de eroare în sistemele de măsurare.
5.2 Aptitudini	A28. Configurează și utilizează sisteme de achiziție a datelor — plăci DAQ, interfețe senzori, lanțuri de măsurare — programând rutine de eșantionare, filtrare și vizualizare a datelor în timp real.
5.3 Responsabilitate și autonomie	R28. Gestionează autonom configurarea și mentenanța sistemelor de achiziție a datelor, asumându-și responsabilitatea pentru calibrarea periodică, documentarea lanțurilor de măsurare și validarea integrității datelor colectate.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Semnale și sisteme de măsurare. Clasificarea semnalelor: analogice, digitale, deterministe, stohastice, periodice, tranzitorii. Lanțul de măsurare: senzor, traductor, condiționare semnal, ADC, prelucrare digitală, afișare/stocare. Caracteristicile metrologice: precizie, exactitate, rezoluție, repetabilitate, liniaritate, histerezis. Erori de măsurare: sistematice, aleatorii, de cuantizare; propagarea erorilor.	2	Prelegere interactivă	
Tema 2: Senzori și transductoare. Senzori de temperatură: termorezistențe (PT100, PT1000 — curba R-T, montajul Wheatstone), termocupluri (efectul Seebeck, compensarea joncțiunii reci, tipuri J/K/T), senzori semiconductori (LM35, DS18B20). Senzori de presiune: piezoresistivi, capacitivi, BMP280. Senzori de deplasare: potențiometrici, inductivi (LVDT), encodere incrementale. Senzori inerțiali: accelerometre MEMS (MPU-6050), giroscopae.	2	Prelegere interactivă	
Tema 3: Condiționarea semnalului. Amplificarea: amplificatoare de instrumentație (INA128), câștig, zgomot referit la intrare (RTI). Filtrarea analogică: filtre trece-jos RC și active (Butterworth, Chebyshev) pentru anti-aliasing. Protecția la supratensiune: limitatoare, optoizolatoare. Conversia curent-tensiune (4-20 mA). Punte Wheatstone: dezechilibru, sensibilitate, compensare termică. Izolarea galvanică.	2	Prelegere interactivă	
Tema 4: Conversia analog-digitală. Principiul ADC: cuantizarea, eroarea de cuantizare,	2	Prelegere interactivă	

rezoluția în biți. Tipuri de ADC: cu aproximații succesive (SAR — cel mai utilizat), sigma-delta (precizie ridicată), flash (viteză mare). Parametrii ADC: rata de eșantionare, ENOB (Effective Number of Bits), SNR, SINAD, THD. DAC: principii, tipuri (R-2R, sigma-delta), parametrii.			
Tema 5: Teorema eșantionării Shannon-Nyquist. Teorema Shannon-Nyquist: frecvența de eșantionare minimă ($f_s \geq 2 \times f_{max}$). Aliasing: fenomenul, efectele, exemplu numeric. Filtrul anti-aliasing: proiectarea pentru o frecvență de tăiere dată și o rată de eșantionare aleasă. Eșantionarea sub-Nyquist: aplicații legitime (bandpass sampling). Reconstituirea semnalului: interpolarea sinc ideală vs. practică. Erori de jitter.	2	Prelegere, demonstrații numerice	
Tema 6: Analiza spectrală. Transformata Fourier Discretă (DFT): definiție, proprietăți, relația cu transformata Fourier continuă. Algoritmul FFT (Fast Fourier Transform): complexitatea $O(N \log N)$ vs. $O(N^2)$. Rezoluția în frecvență: $\Delta f = f_s/N$. Efectul de spectral leakage: cauzele și reducerea prin ferestre (Hanning, Hamming, Blackman, Kaiser). Zero-padding. STFT (Short-Time Fourier Transform) și spectrograma.	2	Prelegere interactivă, MATLAB/Python	
Tema 7: Filtrarea digitală — FIR. Filtrele FIR (Finite Impulse Response): definiție, răspunsul impuls finit, lipsa feedback-ului. Proiectarea FIR prin ferestrea: metoda ferestrei (Hanning, Hamming, Blackman, Kaiser) pentru filtre trece-jos, trece-sus, bandpass, bandstop. Relația între lungimea filtrului (N), frecvența de tăiere și atenuarea benzii de oprire. Faza liniară a filtrelor FIR: avantaje și dezavantaje. Implementarea convoluției.	2	Prelegere interactivă	
Tema 8: Filtrarea digitală — IIR. Filtrele IIR (Infinite Impulse Response): definiție, ecuația cu diferențe, diagrama de flux de semnal. Tipuri de filtre IIR: Butterworth (maximally flat), Chebyshev I și II, Bessel, Elliptic. Proiectarea prin transformare bilineară (metoda Tustin): prewarping-ul frecvențelor. Implementarea în forme canonice directe. Comparatia FIR vs. IIR: ordine, fază, stabilitate, complexitate computațională.	2	Prelegere interactivă	
Tema 9: Filtrarea adaptivă și Kalman. Filtrul Wiener discret: ecuațiile Wiener-Hopf, soluția prin inversarea matricei de corelație. Algoritmul LMS (Least Mean Squares): actualizarea coeficienților, convergența, pulsul de adaptare μ . Algoritmul RLS (Recursive Least Squares): actualizarea recursivă, convergența mai rapidă. Filtrul	2	Prelegere interactivă	

Kalman discret: predicție și actualizare, câștigul Kalman. Aplicarea filtrului Kalman la un accelerometru zgomotos.			
Tema 10: Sisteme DAQ — hardware. Arhitectura unui sistem DAQ: multiplexorul de intrare, circuitul de eșantionare și menținere (S&H), ADC, buffer FIFO, interfața cu calculatorul. Plăci DAQ NI (National Instruments): USB-6001 (8 canale AI, 16 biți, 20 kS/s), PCI-6251 (parametri avansați). Moduri de achiziție: software timing, hardware timing, hardware triggering. Sincronizarea mai multor plăci DAQ.	2	Prelegere interactivă	
Tema 11: Sisteme DAQ — programare Python și LabVIEW. NI-DAQmx Python API: nidaqmx.Task, configurarea canalelor analogice (ai_voltage_chan, ao_voltage_chan), citirea continuă cu callback. Vizualizarea în timp real cu Matplotlib FuncAnimation. Stocarea datelor: HDF5 cu h5py, CSV cu pandas. LabVIEW: introducere în programarea grafică G, VI (Virtual Instrument), DAQ Assistant. Compararea Python vs. LabVIEW pentru sisteme DAQ.	2	Prelegere interactivă, demonstrații cod	
Tema 12: Prelucrarea statistică a datelor de măsurare. Media, varianța, deviația standard, eroarea standard a mediei. Histograma și funcția de distribuție cumulativă. Testul de normalitate (Shapiro-Wilk, Q-Q plot). Identificarea și tratarea valorilor aberante (outliers): criteriul 3σ , testul Grubbs. Incertitudinea de măsurare (GUM — Guide to the Expression of Uncertainty in Measurement): tipurile A și B, combinarea incertitudinilor.	2	Prelegere interactivă	
Tema 13: Calibrarea și validarea sistemelor de măsurare. Curba de calibrare: regresia liniară și polinomială, coeficientul de determinare R^2 . Calibrarea unui senzor de temperatură: procedura, curba de corecție, verificarea cu etalon. Metode de calibrare: cu un punct, cu două puncte, cu N puncte. Certificatele de calibrare: traçabilitate la standardele internaționale (SI). Sisteme de management al calității pentru laborare de măsurare: ISO/IEC 17025.	2	Prelegere interactivă	
Tema 14: Aplicații și recapitulare. Sisteme DAQ industriale: PLC (Programmable Logic Controller) vs. PC-based DAQ. Protocoale industriale: Modbus RTU/TCP, OPC-UA — utilizarea în sisteme SCADA. Instrumentație virtuală: tendințe. IoT și senzori wireless: MQTT pentru transmisia datelor de senzori. Recapitulare generală.	2	Prelegere, recapitulare	

	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> 1. National Instruments, NI-DAQmx Help Documentation — referință completă: https://www.ni.com/docs/en-US/bundle/ni-daqmx/ 2. Proakis, J.G., Manolakis, D.G., Digital Signal Processing: Principles, Algorithms and Applications, 4th ed., Pearson, 2006. 3. Webster, J.G. (ed.), Medical Instrumentation: Application and Design, 4th ed., Wiley, 2010. <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Oppenheim, A.V., Schafer, R.W., Discrete-Time Signal Processing, 3rd ed., Pearson, 2009. 5. JCGM 100:2008 — Evaluation of Measurement Data: Guide to the Expression of Uncertainty in Measurement (GUM). Disponibil la: https://www.bipm.org/en/committees/jc/jcgm/publications <p>Resurse software:</p> <ol style="list-style-type: none"> 6. NI-DAQmx Python Documentation — nidaqmx library: https://nidaqmx-python.readthedocs.io/ 7. SciPy Documentation — prelucrarea semnalelor (scipy.signal): https://docs.scipy.org/doc/scipy/reference/signal.html 8. MathWorks Signal Processing Toolbox — documentație MATLAB: https://www.mathworks.com/help/signal/
--	--

6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Analiza spectrală cu Python. Generarea în Python a unor semnale sinusoidale cu frecvențe date, a unui semnal compus și a unui semnal cu zgomot. Calculul FFT cu numpy.fft. Trasarea spectrului de amplitudini și de faze. Demonstrarea aliasing-ului: eșantionarea unui semnal la frecvențe sub Nyquist. Compararea spectrului cu și fără fereastră Hanning.	2	Lucrare practică Python	
Lucrarea L2: Proiectarea și testarea filtrelor FIR. Proiectarea unui filtru FIR trece-jos cu metoda ferestrei (Hanning) în Python cu scipy.signal.firwin. Verificarea răspunsului în frecvență cu freqz. Aplicarea filtrului pe un semnal zgomotos (semnal sinusoidal + zgomot alb) cu lfilter. Compararea filtrelor de ordine 31, 63 și 127: atenuare vs. latență.	2	Lucrare practică Python	
Lucrarea L3: Proiectarea filtrelor IIR și comparație FIR vs. IIR. Proiectarea unui filtru IIR Butterworth trece-jos cu scipy.signal.butter. Compararea cu un filtru Chebyshev de același ordin: ondulația în banda de trecere. Aplicarea filtrului cu filtfilt (fază zero) vs. lfilter. Compararea ordinului necesar FIR vs. IIR pentru aceeași specificație.	2	Lucrare practică Python	
Lucrarea L4: Achiziția de date cu placă DAQ sau Arduino. Configurarea achiziției de tensiune pe un canal analogic cu NI-DAQmx Python sau cu pyserial pentru Arduino. Achiziția continuă (1000 eșantioane la 10 kS/s). Vizualizarea în timp real cu Matplotlib FuncAnimation. Analiza FFT în timp real pe fereastra glisantă a datelor achiziționate. Stocarea datelor în CSV cu timestamp.	2	Lucrare practică DAQ/Arduino	
Lucrarea L5: Lanț de măsurare — senzor de temperatură. Configurarea unui senzor de temperatură (LM35 sau DS18B20) conectat la placa DAQ sau Arduino. Calibrarea	2	Lucrare practică senzori	

senzorului cu două puncte (0°C apă cu gheață, 100°C apă fiartă). Calculul curbei de calibrare prin regresie liniară. Estimarea incertitudinii de măsurare (tipul A și tipul B). Documentarea lanțului de măsurare.			
Lucrarea L6: Filtrul Kalman — aplicație practică. Implementarea filtrului Kalman discret în Python pentru filtrarea datelor de la un accelerometru simulat sau real (MPU-6050 via I2C/SPI). Configurarea matricelor Q (zgomot de proces) și R (zgomot de măsurare). Compararea estimatei Kalman cu media mobilă și cu filtrul Butterworth pe același semnal. Analiza influenței raportului Q/R.	2	Lucrare practică Python	
Lucrarea L7: Proiect integrat și colocviu. Implementarea unui sistem complet de achiziție și prelucrare: senzor fizic (temperatură, accelerometru sau semnale simulate), achiziție cu DAQ sau Arduino, filtrare adaptivă (Kalman sau FIR), analiza FFT, calibrarea și calculul incertitudinii, vizualizare în timp real, stocare HDF5. Documentarea sistemului: schema bloc a lanțului de măsurare, sursele de eroare, limitele de precizie. Colocviu oral.	2	Proiect integrat, colocviu	
	<p>Bibliografie obligatorie laborator</p> <ol style="list-style-type: none"> 1. NI-DAQmx Python Documentation: https://nidaqmx-python.readthedocs.io/ 2. SciPy Documentation — scipy.signal (filtrare, FFT): https://docs.scipy.org/doc/scipy/reference/signal.html 3. Proakis, J.G., Manolakis, D.G., Digital Signal Processing, 4th ed., Pearson, 2006 — exercițiile de proiectare filtre. <p>Bibliografie complementara laborator</p> <ol style="list-style-type: none"> 4. Oppenheim, A.V., Schafer, R.W., Discrete-Time Signal Processing, 3rd ed., Pearson, 2009. 5. JCGM 100:2008 — Guide to the Expression of Uncertainty in Measurement (GUM): https://www.bipm.org/en/committees/jc/jcgm/publications <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. NI-DAQmx Python — librărie de achiziție date NI: https://nidaqmx-python.readthedocs.io/ 7. SciPy Signal Processing — referință: https://docs.scipy.org/doc/scipy/reference/signal.html 8. PySerial Documentation — comunicare serială cu Arduino: https://pyserial.readthedocs.io/ 9. BIPM GUM — ghidul incertitudinii de măsurare: https://www.bipm.org/en/committees/jc/jcgm/publications 		

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

Achiziția și Prelucrarea Datelor acoperă stratul fizic al oricărui sistem inteligent: transformarea fenomenelor fizice reale (temperatură, presiune, mișcare, sunet) în date digitale exploatabile. Această competență este esențială pentru IoT, sisteme embedded cu senzori, robotică, automatizare industrială și orice aplicație de Machine Learning care lucrează cu date din lumea reală.

Componenta de laborator urmărește formarea capacității practice de a configura un lanț de măsurare complet — de la senzor fizic până la stocare și vizualizare în timp real — și de a documenta transparent sursele de eroare și limitele de precizie, direct corelate cu A28 și R28 din matricea de corelare. Proiectul integrat din L7 reproduce un sistem DAQ real de dimensiune redusă.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea principiilor de funcționare ale senzorilor uzuali (temperatură, presiune, inerțiali) și ale ADC; aplicarea corectă a teoremei Shannon-Nyquist și identificarea aliasing-ului; cunoașterea și compararea filtrelor FIR și IIR (Butterworth, Chebyshev); cunoașterea principiilor filtrului Kalman; cunoașterea metodologiei de calibrare și a conceptului de incertitudine de măsurare (GUM).	Examen scris (2 ore): 1 problemă de calcul al frecvenței de eșantionare minime și al filtrului anti-aliasing pentru un semnal dat (cu justificarea alegerii), 1 problemă de proiectare a unui filtru FIR sau IIR (specificarea ordinului, a frecvenței de tăiere și a atenuării benzii de oprire) și 1 problemă de calibrare (calculul curbei de corecție prin regresie liniară și estimarea incertitudinii combinate)	60%
8.5 Seminar/laborator	Funcționalitatea codului Python din laborator (filtrele produc răspunsul în frecvență corect, FFT afișează frecvențele corecte, achiziția generează date valide); calitatea documentației lanțului de măsurare din L5 (schema bloc completă, sursele de eroare identificate, incertitudinea calculată); calitatea proiectului integrat din L7; calitatea prezentării la colocviu.	Evaluare continuă: cod Python și rapoarte predate pe Sakai după fiecare lucrare (50%) și proiect integrat cu colocviu oral individual la L7 (50%)	40%
8.6 Standard minim de performanță:			
Prezentarea la colocviul oral din L7 a schemei bloc complete a lanțului de măsurare implementat — cu identificarea și cuantificarea a cel puțin două surse de eroare (ex: eroarea de cuantizare a ADC și eroarea de neliniaritate a senzorului) și calculul incertitudinii combinate conform GUM. La examenul scris: calculul corect al frecvenței de eșantionare minime și justificarea alegerii frecvenței filtrului anti-aliasing — aceste două verificând înțelegerea teoremei Shannon-Nyquist, conceptul central al disciplinei.			

Data completării 19.03.2026	Semnătura titularului de curs Conf. Dr. Răileanu Alina-Beatrice	Semnătura titularului de seminar / laborator Lect. Dr. Anghelache Iulia
Data avizării în Consiliul Departamentului 24.03.2026	Semnătura directorului DIT Lect. Dr. Antohe Valerian	
Data aprobării în Consiliul Facultății 07.04.2026	Semnătura decanului FMDT Conf. Dr. Beteringhe Adrian	



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei	BAZE DE DATE – TIDS305						
1.2 Titularul activităților de curs	Lect. Dr. Lițan Daniela Elena						
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator	Lect. Dr. Lițan Daniela Elena						
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	E	1.8 Regimul disciplinei	Ob DS

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	4	din care: 2.2 curs	3	2.3 seminar/laborator	0/1
2.4 Total ore din planul de învățământ	56	din care: 2.5 curs	42	2.6 seminar/laborator	0/14
Distribuția fondului de timp					ore
2.7 Activități monitorizate					-
Practică					-
Elaborare proiect diplomă					-
2.8 Activități individuale					44
Documentare					10
Studiu individual					15
Referate					15
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					4
2.9 Total ore pe semestru					100
2.10 Numărul de credite					4

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Structuri de Date și Algoritmi (TIDS203) — liste, arbori, grafuri, complexitatea algoritmilor; Matematici Discrete (TIDS205) — teoria mulțimilor, logică predicativă, relații; Programare Orientată pe Obiecte (TIDS204) — clase, obiecte, moștenire (pentru ORM și mapping).
3.2 de competențe	Cunoașterea conceptelor de bază ale modelului relațional: tabele, chei primare, chei externe, relații 1:1, 1:N, M:N; capacitatea de a scrie interogări SQL de bază (SELECT, INSERT, UPDATE, DELETE, JOIN simplu); familiarizarea cu cel puțin un SGBD (MySQL, PostgreSQL sau SQLite).

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu tablă, videoproiector și acces la internet; platforma Sakai; recomandare: PostgreSQL sau MySQL instalat și pgAdmin sau MySQL Workbench configurate înainte de curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator de informatică cu calculatoare individuale; PostgreSQL și pgAdmin sau MySQL și MySQL Workbench instalate; studenții pot

folosi propriul laptop; platformele online db-fiddle.com sau sqlfiddle.com ca alternativă pentru exerciții rapide.

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C13. Posedă cunoștințe specializate privind proiectarea și administrarea bazelor de date relaționale și avansate, inclusiv tehnicile de optimizare a interogărilor și principiile sistemelor avansate de gestiune a bazelor de date (SGBD).
5.2 Aptitudini	A13. Proiectează și implementează baze de date relaționale și avansate, elaborând scheme de date optimizate, interogări complexe și proceduri de administrare care asigură integritatea, disponibilitatea și securitatea datelor.
5.3 Responsabilitate și autonomie	R13. Contribuie activ la dezvoltarea cunoștințelor și practicilor profesionale din domeniul tehnologiei informației prin documentarea riguroasă a soluțiilor tehnice implementate, elaborând rapoarte, specificații și studii de caz reutilizabile de către comunitatea profesională.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Modelul relațional — fundamente. Istoricul SGBD: de la fișiere plate la modelul relațional (Codd, 1970). Conceptele de bază: relație, tuplu, atribut, domeniu. Chei: primară, candidat, alternativă, externă. Integritatea datelor: constrângeri de entitate, referențială, de domeniu. Algebra relațională: selecție, proiecție, produs cartezian, join natural, reuniune, intersecție, diferență.	3	Prelegere interactivă	
Tema 2: SQL — DDL și DML. DDL: CREATE TABLE cu tipuri de date (VARCHAR, INTEGER, NUMERIC, DATE, BOOLEAN, TEXT), constrângeri (PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, CHECK, DEFAULT). ALTER TABLE, DROP TABLE. DML: INSERT (single și multi-row), UPDATE cu subinterogări, DELETE. TRUNCATE vs. DELETE. Tranzacțiile: BEGIN, COMMIT, ROLLBACK. Nivelele de izolare a tranzacțiilor.	3	Prelegere interactivă, SQL live	
Tema 3: SQL — interogări avansate. SELECT complet: WHERE cu operatori (BETWEEN, IN, LIKE, IS NULL), ORDER BY, LIMIT/OFFSET. JOIN-uri: INNER JOIN, LEFT/RIGHT OUTER JOIN, FULL OUTER JOIN, CROSS JOIN, SELF JOIN. Funcții de agregare: COUNT, SUM, AVG, MIN, MAX. GROUP BY și HAVING. Subinterogări: scalare, de tip listă (IN, ANY, ALL), corelate (EXISTS). UNION, INTERSECT, EXCEPT.	3	Prelegere interactivă, SQL live	
Tema 4: Funcții, view-uri și proceduri stocate. Funcții scalare predefinite: string (UPPER, LOWER, TRIM, SUBSTRING, CONCAT), numerice (ROUND, ABS, MOD), de dată (NOW, DATE PART, AGE). Funcții window	3	Prelegere interactivă, SQL live	

(analitice): ROW_NUMBER(), RANK(), DENSE_RANK(), LAG(), LEAD(), PARTITION BY. VIEW-uri: simple și complexe, WITH CHECK OPTION. Proceduri stocate și funcții definite de utilizator: sintaxa PL/pgSQL (PostgreSQL) sau T-SQL (SQL Server).			
Tema 5: Normalizarea bazelor de date. Anomaliile de actualizare, inserare și ștergere. Dependentele funcționale: definiție, closure, acoperire minimală. Forma normală 1 (1NF): eliminarea grupurilor repetitive. 2NF: eliminarea dependențelor parțiale față de cheia primară. 3NF: eliminarea dependențelor tranzitive. BCNF (Boyce-Codd Normal Form): definiție și comparație cu 3NF. 4NF: dependențele multivaloare. Denormalizarea: motive și tehnici.	3	Prelegere, demonstrații	
Tema 6: Modelarea entitate-relație (ER). Diagrama ER: entitate, atribut, relație, cardinalitate (1:1, 1:N, M:N), participare totală/parțială. Entități slabe. Atribute compuse, multivaloare, derivate. Notățiile: Chen, Crow's Foot, UML. Conversia diagramei ER la schema relațională: regulile de transformare pentru fiecare tip de relație. Modelarea extinsă (EER): specializare, generalizare, agregare.	3	Prelegere, exerciții modelare	
Tema 7: Indexarea și optimizarea interogărilor. Structuri de index: B-tree (cel mai utilizat), Hash, GiST, GIN (pentru text full-text). Indexuri compuse, parțiale și acoperitoare (covering index). Planul de execuție al interogărilor: EXPLAIN și EXPLAIN ANALYZE în PostgreSQL. Statisticile optimizerului: pg_stats, pg_class. Tehnici de optimizare: alegerea tipului corect de JOIN, evitarea funcțiilor pe coloanele indexate în WHERE, partitionarea.	3	Prelegere interactivă	
Tema 8: Tranzacții și concurența. Proprietățile ACID: atomicitate, consistență, izolare, durabilitate. Probleme de concurență: dirty read, non-repeatable read, phantom read. Nivelele de izolare ISO: READ UNCOMMITTED, READ COMMITTED, REPEATABLE READ, SERIALIZABLE. Mecanisme de control al concurenței: blocare pesimistă (shared/exclusive locks), blocare optimistă (MVCC — Multi-Version Concurrency Control, implementat în PostgreSQL). Deadlock: detecție și rezolvare.	3	Prelegere interactivă	
Tema 9: Administrarea SGBD. Arhitectura PostgreSQL: procese (postmaster, backend, bgwriter), structura fișierelor. Crearea și gestionarea utilizatorilor și a rolurilor. Privilegii: GRANT, REVOKE, Row-Level	3	Prelegere interactivă	

Security (RLS). Backup și restaurare: pg_dump, pg_dumpall, pg_restore, PITR (Point-In-Time Recovery) cu WAL. Monitorizarea performanței: pg_stat_activity, pg_stat_statements, autovacuum.			
Tema 10: Baze de date distribuite. Conceptele de distribuție: transparența locației, fragmentarea (orizontală, verticală, mixtă), replicarea. Teorema CAP: consistență, disponibilitate, toleranță la partiționare — alegeri de proiectare. Consistența eventuală (eventual consistency). PostgreSQL și distribuția: logical replication, streaming replication. Sharding: principii, consistența distribuită (two-phase commit). CockroachDB și YugabyteDB: arhitecturi SQL distribuite.	3	Prelegere interactivă	
Tema 11: Baze de date NoSQL. Motivațiile NoSQL: scalabilitate orizontală, schemă flexibilă, Big Data. Tipurile de baze de date NoSQL: document (MongoDB), cheie-valoare (Redis), coloană-largă (Apache Cassandra), graf (Neo4j). MongoDB: modelul de document BSON, colecții, operații CRUD, indexuri, aggregation pipeline. Redis: structuri de date (string, hash, list, set, sorted set), persistența (RDB/AOF), cazuri de utilizare (cache, session store, pub/sub).	3	Prelegere interactivă	
Tema 12: Baze de date în cloud și Big Data. Servicii de baze de date cloud: AWS RDS/Aurora, Azure SQL Database, Google Cloud SQL — SaaS vs. PaaS, avantaje și limitări față de on-premise. Serverless databases: Aurora Serverless, PlanetScale. Data Warehouse: arhitectura (OLTP vs. OLAP), modelul dimensional (stea, fulg de zăpadă), Amazon Redshift, Google BigQuery. Apache Hadoop și Spark SQL: procesarea datelor la scară mare. Lakehouse: Delta Lake, Apache Iceberg.	3	Prelegere interactivă	
Tema 13: Securitatea bazelor de date. Modelul de securitate: autentificare, autorizare, auditare. Atacul SQL Injection: tipuri (clasic, blind, time-based), exploatarea, prevenirea prin interogări parametrizate (prepared statements) și ORM. Criptarea datelor: at-rest (Transparent Data Encryption — TDE) și in-transit (SSL/TLS). Mascarea datelor (data masking). Conformitatea: GDPR — dreptul la ștergere, pseudonimizarea, rolul DBA în conformitate.	3	Prelegere interactivă	
Tema 14: ORM și tendințe — recapitulare. ORM (Object-Relational Mapping): principii, avantaje și dezavantaje față de SQL raw. SQLAlchemy (Python): declarative mapping, session, relații (relationship), lazy vs. eager loading, generarea automată a schemei.	3	Prelegere, recapitulare	

<p>Hibernate (Java) și Entity Framework (.NET). NewSQL: bazele de date care combină consistența ACID cu scalabilitatea orizontală. Recapitulare generală.</p>			
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> 1. Ramakrishnan, R., Gehrke, J., Database Management Systems, 3rd ed., McGraw-Hill, 2002. 2. Silberschatz, A., Korth, H.F., Sudarshan, S., Database System Concepts, 7th ed., McGraw-Hill, 2019. 3. PostgreSQL Documentation — referință completă a SGBD PostgreSQL: https://www.postgresql.org/docs/ <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Kleppmann, M., Designing Data-Intensive Applications, O'Reilly Media, 2017. 5. Date, C.J., An Introduction to Database Systems, 8th ed., Addison-Wesley, 2003. <p>Resurse software:</p> <ol style="list-style-type: none"> 6. PostgreSQL Documentation — manual complet: https://www.postgresql.org/docs/ 7. MySQL Documentation — referință SQL și administrare: https://dev.mysql.com/doc/ 8. db-fiddle.com — mediu online pentru testarea interogărilor SQL: https://www.db-fiddle.com/ 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
<p>Lucrarea L1: Modelarea și crearea schemei. Proiectarea diagramei ER pentru o temă la alegere (bibliotecă online, spital, platformă e-learning). Conversia diagramei ER la schema relațională. Crearea schemei în PostgreSQL cu DDL complet: tipuri de date corecte, toate constrângerile (PK, FK, UNIQUE, CHECK, NOT NULL). Inserarea unui set de date de test cu INSERT.</p>	2	Lucrare practică PostgreSQL	
<p>Lucrarea L2: Interogări SQL — JOIN-uri și agregare. Scrierea a minim 10 interogări SELECT de complexitate crescătoare pe schema creată în L1: INNER JOIN pe 3 tabele, LEFT JOIN pentru raportarea valorilor lipsă, GROUP BY cu HAVING pentru filtrare pe agregate, subinterogări corelate cu EXISTS. Compararea planurilor de execuție (EXPLAIN) pentru 2 variante ale aceleiași interogări.</p>	2	Lucrare practică PostgreSQL	
<p>Lucrarea L3: Funcții window și view-uri. Implementarea a 5 interogări cu funcții window: ROW_NUMBER() pentru paginare, RANK() pentru clasamente, LAG()/LEAD() pentru comparații cu înregistrări adiacente. Crearea a 2 view-uri: unul simplu (raport) și unul complex (cu JOIN și agregare). Crearea unei funcții definite de utilizator în PL/pgSQL pentru o logică de business specifică schemei.</p>	2	Lucrare practică PostgreSQL	
<p>Lucrarea L4: Normalizarea — exerciții aplicate. Analiza unei scheme denormalizate (furnizată) cu identificarea tuturor anomaliilor și a dependențelor funcționale. Aducerea la 3NF prin pași documentați (1NF → 2NF → 3NF). Implementarea schemei normalizate în PostgreSQL. Compararea volumului de date și a performanței interogărilor de citire vs. scriere față de schema originală.</p>	2	Lucrare practică PostgreSQL	
<p>Lucrarea L5: Indexare și optimizare. Crearea unui set de date mare (minim 100.000</p>	2	Lucrare practică PostgreSQL	

înregistrări) cu <code>generate_series()</code> sau <code>pg_bench</code> . Măsurarea timpului de execuție al 5 interogări fără indecși. Adăugarea de indecși B-tree simpli, compuși și parțiali. Remăsurarea și compararea planurilor EXPLAIN ANALYZE. Identificarea cazurilor în care indexul nu este folosit de optimizer.			
Lucrarea L6: Tranzacții, securitate și backup. Demonstrarea comportamentului ACID: un scenariu de transfer bancar cu BEGIN/COMMIT/ROLLBACK. Demonstrarea unui deadlock și rezolvarea sa. Crearea de roluri și utilizatori cu privilegii minime necesare (principiul least privilege). Backup cu <code>pg_dump</code> și restaurarea bazei de date. Demonstrarea unui atac SQL Injection pe o interogare vulnerabilă și prevenirea cu prepared statements.	2	Lucrare practică PostgreSQL	
Lucrarea L7: Proiect integrat și colocviu. Proiectarea și implementarea completă a unei baze de date relaționale pentru un scenariu dat: diagrama ER, schema DDL normalizată la 3NF, set de date realiste, minim 15 interogări SQL de complexitate variată (inclusiv funcții window și subinterogări corelate), indecși justificați, cel puțin un rol cu privilegii restricționate și documentarea soluției. Colocviu oral.	2	Proiect individual, colocviu	
	<p>Bibliografie obligatorie laborator</p> <ol style="list-style-type: none"> 1. PostgreSQL Documentation — DDL, DML, PL/pgSQL, EXPLAIN: https://www.postgresql.org/docs/ 2. Ramakrishnan, R., Gehrke, J., Database Management Systems, 3rd ed., McGraw-Hill, 2002 — exercițiile de normalizare și interogări. 3. db-fiddle.com — testarea interogărilor online: https://www.db-fiddle.com/ <p>Bibliografie complementara laborator</p> <ol style="list-style-type: none"> 4. Kleppmann, M., Designing Data-Intensive Applications, O'Reilly Media, 2017 — capitolele de tranzacții și baze de date distribuite. 5. pgAdmin Documentation — ghid de utilizare pgAdmin: https://www.pgadmin.org/docs/ <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. PostgreSQL Documentation — referință completă: https://www.postgresql.org/docs/ 7. SQLZoo — tutoriale SQL interactive: https://sqlzoo.net/ 8. pgAdmin — interfață grafică pentru PostgreSQL: https://www.pgadmin.org/ 9. db-fiddle.com — mediu SQL online: https://www.db-fiddle.com/ 		

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

Bazele de Date furnizează competențele esențiale pentru orice rol în industria software care implică persistența datelor: de la proiectarea schemelor relaționale normalizate și scrierea interogărilor complexe, până la administrarea SGBD-urilor, optimizarea performanței și securizarea datelor. Practic orice aplicație web, mobilă sau enterprise stochează date într-un SGBD, iar calitatea proiectării bazei de date determină direct calitatea și scalabilitatea întregii aplicații. Acoperirea NoSQL, a bazelor de date cloud și a tehnologiilor Big Data (Temele 11-12) asigură studentului perspectiva comparativă necesară alegerii soluției potrivite pentru fiecare context, conform C13. Documentarea riguroasă a proiectului din L7 — diagramă ER, justificarea normalizării, indexurile justificate — cultivă direct competența R13: elaborarea de specificații și studii de caz reutilizabile.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea și aplicarea algebrei relaționale și a normalizării (1NF-BCNF) cu identificarea dependențelor funcționale; scrierea interogărilor SQL complexe (JOIN-uri multiple, subinterogări corelate, funcții window, GROUP BY cu HAVING); cunoașterea tehnicilor de indexare și interpretarea planului de execuție EXPLAIN; cunoașterea proprietăților ACID și a nivelelor de izolare; cunoașterea comparativă a bazelor de date NoSQL și cloud.	Examen scris (2 ore): 1 problemă de modelare ER cu conversia la schema relațională, 1 problemă de normalizare (identificarea dependențelor funcționale și aducerea la 3NF documentat pas cu pas) și 1 problemă de interogări SQL (3 interogări de complexitate crescătoare pe o schemă dată, inclusiv o interogare cu funcție window sau subinterogare corelată)	40%
8.5 Seminar/laborator	Corectitudinea schemei DDL din proiectul de laborator (constrângeri corecte și complete, normalizare la 3NF justificată); corectitudinea și complexitatea interogărilor SQL (cel puțin una cu funcție window și una cu subinterogare corelată); calitatea documentației proiectului (diagrama ER, justificarea indecșilor, specificațiile de securitate); calitatea prezentării la colocviu.	Evaluare continuă: livrabile individuale predate pe Sakai după fiecare lucrare (50%) și proiect integrat cu colocviu oral individual la L7 (50%)	60%
8.6 Standard minim de performanță:			
Predarea unui proiect de baze de date care conține obligatoriu o schemă DDL normalizată la 3NF cu justificarea pașilor de normalizare — identificarea dependențelor funcționale, anomaliile eliminate și tabelele rezultate — ȘI cel puțin o interogare SQL cu funcție window (ROW_NUMBER, RANK sau echivalent). Schema fără constrângeri de integritate referențială sau proiectul fără documentarea normalizării nu îndeplinesc standardul minim, indiferent de complexitatea interogărilor.			

Data completării 18.03.2026	Semnătura titularului de curs Lect. Dr. Lițan Daniela Elena	Semnătura titularului de seminar / laborator Lect. Dr. Lițan Daniela Elena
Data avizării în Consiliul Departamentului 24.03.2026	Semnătura directorului DIT Lect. Dr. Antohe Valerian	
Data aprobării în Consiliul Facultății 07.04.2026	Semnătura decanului FMDT Conf. Dr. Beteringhe Adrian	



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei		BAZE DE DATE - PROIECT– TIDS306					
1.2 Titularul activităților de curs							
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator / proiect		Lect. Dr. Lițan Daniela -Elena					
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	P	1.8 Regimul disciplinei	Ob DS

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	1	din care: 2.2 curs		2.3 seminar/laborator/proiect	0/0/1
2.4 Total ore din planul de învățământ	14	din care: 2.5 curs		2.6 seminar/laborator/proiect	0/0/14
Distribuția fondului de timp					ore
2.7 Activități monitorizate					
Practică					
Elaborare proiect diplomă					
2.8 Activități individuale					
Documentare					
Studiu individual					
Referate					
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					
2.9 Total ore pe semestru					25
2.10 Numărul de credite					1

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Baze de Date (TIDS305) — modelul relațional, SQL complet, normalizare, indexare, tranzații, NoSQL, baze de date cloud; Programare Orientată pe Obiecte (TIDS204) — clase, obiecte (pentru nivelul de acces la date al proiectului).
3.2 de competențe	Capacitatea de a proiecta o schemă relațională normalizată la 3NF; cunoașterea SQL-ului la nivel mediu-avansat (JOIN-uri, subinterogări, funcții de agregare); familiarizarea cu pgAdmin sau MySQL Workbench; abilitatea de a redacta documentație tehnică structurată.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	
4.2 de desfășurare a seminarului/laboratorului	Activitatea de proiect se desfășoară individual, la calculatorul propriu sau în laboratorul de informatică; PostgreSQL și pgAdmin sau MySQL și MySQL Workbench instalate; întâlnirile de îndrumare cu titularul au loc săptămânal, față

în față sau online, cu prezentarea stadiului și primirea feedback-ului; predarea finală pe platforma Sakai.

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C13. Posedă cunoștințe specializate privind proiectarea și administrarea bazelor de date relaționale și avansate, inclusiv tehnicile de optimizare a interogărilor și principiile sistemelor avansate de gestiune a bazelor de date (SGBD). C14. Cunoaște și aplică critic principiile arhitecturii datelor în medii cloud și distribuite, cu conștientizarea tendințelor actuale în domeniul big data și al bazelor de date NoSQL.
5.2 Aptitudini	A13. Proiectează și implementează baze de date relaționale și avansate, elaborând scheme de date optimizate, interogări complexe și proceduri de administrare care asigură integritatea, disponibilitatea și securitatea datelor.
5.3 Responsabilitate și autonomie	R13. Contribuie activ la dezvoltarea cunoștințelor și practicilor profesionale din domeniul tehnologiei informației prin documentarea riguroasă a soluțiilor tehnice implementate, elaborând rapoarte, specificații și studii de caz reutilizabile de către comunitatea profesională.

6. Conținuturi

6.1 Proiect	Nr. ore	Metode de predare	Observații
P1: Alegerea temei și definirea cerințelor. Prezentarea categoriilor de proiecte disponibile: platforme e-commerce, sisteme de gestiune (bibliotecă, spital, universitate, hotel), aplicații analitice. Alegerea temei individuale. Identificarea entităților principale, a relațiilor și a cerințelor funcționale. Redactarea documentului de cerințe (1-2 pagini).	1	Îndrumare individuală	
P2: Diagrama ER — versiunea 1. Elaborarea diagramei ER inițiale: entități, atribute, relații cu cardinalitate. Identificarea cheilor primare pentru fiecare entitate. Discutarea entităților slabe și a relațiilor ternare dacă există. Feedback din partea titularului pe corectitudinea modelului.	1	Îndrumare, feedback diagramă	
P3: Diagrama ER — versiunea finală. Revizuirea diagramei pe baza feedback-ului. Completarea atributelor (compuse, multivaloare, derivate). Specificarea participării totale/parțiale. Finalizarea diagramei ER în notație Crow's Foot sau Chen, cu toate cardinalitățile.	1	Îndrumare, feedback	
P4: Conversia ER la schema relațională. Aplicarea regulilor de conversie pentru toate tipurile de relații (1:1, 1:N, M:N). Identificarea tuturor cheilor externe. Verificarea că schema reflectă fidel diagrama ER. Documentarea deciziilor de conversie pentru cazurile ambigue.	1	Îndrumare, validare schemă	
P5: Normalizarea schemei. Analiza dependențelor funcționale pentru fiecare tabelă. Verificarea 1NF, 2NF, 3NF —	1	Îndrumare, validare normalizare	

identificarea și eliminarea anomaliilor. Documentarea pașilor de normalizare: tabele înainte și după, cu justificarea fiecărei transformări.			
P6: Implementarea DDL. Scrierea și executarea scriptului DDL complet în PostgreSQL sau MySQL: CREATE TABLE cu toate constrângerile (PK, FK, UNIQUE, NOT NULL, CHECK, DEFAULT). Verificarea că toate relațiile și constrângerile sunt implementate corect. Testul de inserare a câtorva înregistrări.	1	Implementare SQL, feedback	
P7: Date de test și interogări de bază. Popularea bazei de date cu un set de date realiste (minim 50-100 înregistrări per tabelă principală), manual sau cu script. Scrierea primelor 5 interogări SELECT cu JOIN-uri pe cel puțin 3 tabele. Prezentare intermediară: diagrama ER, schema DDL, date de test.	1	Implementare, prezentare intermediară	
P8: Interogări avansate. Implementarea a minim 5 interogări avansate: subinterogări corelate (cu EXISTS sau IN), interogări cu GROUP BY și HAVING, subinterogări în clauza FROM. Testarea și verificarea corectitudinii rezultatelor față de datele de test.	1	Implementare SQL	
P9: Funcții window și view-uri. Implementarea a minim 3 interogări cu funcții window (ROW_NUMBER, RANK, LAG sau LEAD). Crearea a minim 2 view-uri: unul pentru raportare, unul pentru simplificarea interogărilor complexe. Testarea view-urilor cu interogări SELECT.	1	Implementare SQL	
P10: Optimizare și indexare. Analiza planului de execuție (EXPLAIN ANALYZE) pentru cele mai costisitoare interogări ale proiectului. Adăugarea indexurilor justificate (B-tree simplu sau compus). Documentarea: planul de execuție înainte și după index, cu comentariul motivației.	1	Implementare SQL, analiză EXPLAIN	
P11: Proceduri stocate și triggere. Implementarea a cel puțin unei proceduri stocate sau funcții PL/pgSQL pentru o logică de business identificată în proiect. Opțional: un trigger pentru menținerea integrității sau pentru auditare (ex: log al modificărilor). Testarea cu scenarii variate.	1	Implementare PL/pgSQL	
P12: Securitate și backup. Crearea unui rol cu privilegii minime necesare aplicației (SELECT pe tabele de citire, INSERT/UPDATE pe tabele de scriere). Demonstrarea că rolul nu poate accesa tabelele restricționate. Backup al bazei de date cu pg_dump și verificarea restaurării pe o bază de date de test.	1	Implementare SQL, backup	
P13: Redactarea documentației finale. Structura documentației: introducere, cerințe,	1	Redactare tehnică	

diagrama ER, schema relațională (DDL), normalizarea (cu justificare), interogările reprezentative (cu explicarea logicii), indexurile (cu justificare), concluzii. Inserarea capturilor de ecran relevante.			
P14: Prezentarea și evaluarea proiectului final. Prezentarea orală (10 minute + 5 minute Q&A): demonstrarea live a bazei de date, explicarea deciziilor de proiectare (normalizare, indexuri, securitate), rularea interogărilor reprezentative. Predarea documentației și a scripturilor SQL pe Sakai.	1	Prezentare orală, evaluare	
	<p>Bibliografie obligatorie proiect</p> <ol style="list-style-type: none"> Ramakrishnan, R., Gehrke, J., Database Management Systems, 3rd ed., McGraw-Hill, 2002. Silberschatz, A., Korth, H.F., Sudarshan, S., Database System Concepts, 7th ed., McGraw-Hill, 2019. PostgreSQL Documentation — DDL, PL/pgSQL, EXPLAIN: https://www.postgresql.org/docs/ <p>Bibliografie complementara proiect</p> <ol style="list-style-type: none"> Kleppmann, M., Designing Data-Intensive Applications, O'Reilly Media, 2017. pgAdmin Documentation — ghid de utilizare: https://www.pgadmin.org/docs/ <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> PostgreSQL Documentation — referință completă: https://www.postgresql.org/docs/ pgAdmin — interfață grafică PostgreSQL: https://www.pgadmin.org/ SQLZoo — tutoriale SQL interactive pentru recapitulare: https://sqlzoo.net/ db-fiddle.com — testarea rapidă a interogărilor online: https://www.db-fiddle.com/ 		

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

<p>Proiectul de Baze de Date integrează competențele din disciplina omonimă (TIDS305) într-un produs complet: de la modelarea ER și normalizare până la implementarea DDL, interogările avansate, optimizarea și documentarea. Studentul parcurge autonom întregul ciclu de proiectare a unei baze de date, reproducând contextul unui proiect real din industrie.</p> <p>Documentarea riguroasă cerută la fiecare etapă — justificarea normalizării, motivarea indexurilor, specificațiile de securitate — cultivă direct competența R13: elaborarea de rapoarte și specificații tehnice reutilizabile. Prezentarea orală din P14 verifică capacitatea de a justifica deciziile de proiectare în fața unui evaluator, competență esențială în orice rol de arhitect sau administrator de date.</p>

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs			
8.5 seminar/laborator/proiect	Funcționalitatea bazei de date demonstrate live la P14 (baza de date pomește autonom, constrângerile sunt active, interogările produc rezultatele corecte); calitatea și completitudinea documentației (diagrama ER, pașii de normalizare cu	Proiect individual evaluat la P14: demonstrație live (40%) + documentație predată pe Sakai (40%) + prezentare orală cu Q&A de 15 minute (20%)	100%

	justificare, indexurile motivate, rolul de securitate); calitatea prezentării orale și a răspunsurilor la întrebări.		
--	--	--	--

8.6 Standard minim de performanță:

Predarea documentației care conține obligatoriu diagrama ER finală ȘI pașii de normalizare documentați (dependențele funcționale identificate, anomaliile eliminate, tabelele înainte și după) — aceasta diferențiind proiectul de o simplă colecție de tabele. La demonstrația live din P14: baza de date trebuie să pornească fără intervenție manuală și cel puțin o interogare cu funcție window să producă rezultatele corecte, dovedind că studentul a înțeles nu doar să scrie SQL, ci și să justifice deciziile de proiectare.

Data completării	Semnătura titularului de curs	Semnătura titularului de seminar / laborator/proiect Lect. Dr. Lițan Daniela-Elena
19.03.2026		
Data avizării în Consiliul Departamentului	Semnătura directorului DIT Lect. dr. Antohe Valerian	
24.03.2026		
Data aprobării în Consiliul Facultății	Semnătura decanului FMDT Conf. dr. Beteringhe Adrian	
07.04.2026		



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei	PROIECTAREA ALGORITMILOR – TIDS307						
1.2 Titularul activităților de curs	Lect. Dr. Panfiloiu Gheorghe						
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator	Lect. Dr. Panfiloiu Gheorghe						
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	E	1.8 Regimul disciplinei	Ob DS

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	3	din care: 2.2 curs	2	2.3 seminar/laborator	0/1
2.4 Total ore din planul de învățământ	42	din care: 2.5 curs	28	2.6 seminar/laborator	0/14
Distribuția fondului de timp					ore
2.7 Activități monitorizate					-
Practică					-
Elaborare proiect diplomă					-
2.8 Activități individuale					58
Documentare					18
Studiu individual					18
Referate					18
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					4
2.9 Total ore pe semestru					100
2.10 Numărul de credite					4

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Structuri de Date și Algoritmi (TIDS203) — liste, stive, cozi, arbori, grafuri, sortare, căutare, complexitate O; Matematici Discrete (TIDS205) — combinatorică, teoria grafurilor, recurențe; Programare Orientată pe Obiecte (TIDS204) — implementarea structurilor de date în C++ sau Java.
3.2 de competențe	Cunoașterea și implementarea structurilor de date fundamentale (liste, arbori binari, heap, hash table, grafuri); analiza complexității temporale și spațiale cu notația O mare; implementarea algoritmilor clasici de sortare și căutare; rezolvarea recurențelor simple prin substituție sau teorema master.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu tablă pentru demonstrații matematice, videoproiector și acces la internet; platforma Sakai; recomandare: un compilator C++ sau Java și acces la platforma online LeetCode sau Codeforces pentru experimentarea algoritmilor prezentați.
4.2 de desfășurare a seminarului/laboratorului	Laborator de informatică cu calculatoare individuale; compilator C++ (GCC) sau Java (JDK); studenții pot folosi propriul laptop; acces la

platformele LeetCode și Codeforces pentru submisia și testarea automată a soluțiilor; platforma Sakai pentru predarea implementărilor.

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C8. Cunoaște în profunzime structurile de date și algoritmi fundamentali, inclusiv algoritmi paraleli și distribuți, cu capacitatea de a analiza critic complexitatea și eficiența soluțiilor propuse.
5.2 Aptitudini	A6. Proiectează și analizează algoritmi eficienți pentru rezolvarea problemelor computaționale complexe, evaluând critic complexitatea temporală și spațială a soluțiilor propuse și alegând structurile de date adecvate.
5.3 Responsabilitate și autonomie	R8. Răspunde în mod responsabil și transparent față de echipă și beneficiari pentru respectarea termenelor, a bugetului tehnic și a specificațiilor agreeate, comunicând proactiv orice deviere sau risc identificat.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Analiza complexității algoritmilor. Recapitulare notații asimptotice: O , Ω , Θ . Analiza cazului defavorabil, mediu și favorabil. Rezolvarea recurențelor: metoda substituției, arborele de recursivitate, teorema master (toate cele 3 cazuri). Amortizarea: metoda agregată, a potențialului. Exemple: tablourile dinamice (std::vector), structuri disjuncte union-find.	2	Prelegere interactivă	
Tema 2: Divide et Impera. Paradigma divide et impera: împărțire, rezolvare recursivă, combinare. Merge Sort: analiza $T(n)=2T(n/2)+O(n)$, stabilitate. Quick Sort: alegerea pivotului (random, median-of-3), analiza cazului mediu $O(n \log n)$. Algoritmul lui Strassen pentru înmulțirea matricelor: $T(n)=7T(n/2)+O(n^2)$. Binary Search generalizat: căutarea în spații de soluții necontinue.	2	Prelegere, analiza complexității	
Tema 3: Programare dinamică — fundamente. Principiul optimalității Bellman. Suprapunerea subproblemelor și substructura optimă. Memoizare (top-down) vs. tabulare (bottom-up). Problema rucsacului 0/1: formularea recursivă, tabela DP, reconstruirea soluției, complexitatea $O(nW)$. Longest Common Subsequence (LCS): matricea DP, reconstruirea LCS, complexitate $O(mn)$.	2	Prelegere, demonstrații	
Tema 4: Programare dinamică — aplicații avansate. Înmulțirea optimă a lanțului de matrice (Matrix Chain Multiplication): $O(n^3)$. Edit Distance (distanța Levenshtein): aplicații în bioinformatică și NLP. Longest Increasing Subsequence (LIS): soluția $O(n^2)$ și soluția optimă $O(n \log n)$ cu căutare binară. Problema intervalelor ponderate (Weighted Job	2	Prelegere, demonstrații	

Scheduling). DP pe arbori: independent set maximal.			
Tema 5: Algoritmi greedy. Proprietatea alegerii greedy și substructura optimă: condiții de corectitudine. Problema selecției activităților (Activity Selection): proba de corectitudine. Coduri Huffman: construcția arborelui cu heap de minime, complexitate $O(n \log n)$, demonstrarea optimalității. Algoritmul lui Kruskal pentru MST: Union-Find, complexitate $O(E \log E)$. Algoritmul lui Prim: cu heap binar $O((V+E) \log V)$.	2	Prelegere interactivă	
Tema 6: Algoritmi pe grafuri — parcurgere și componente. BFS: implementarea cu coadă, proprietățile arborelui BFS, cel mai scurt drum în grafuri neponderate. DFS: implementarea cu stivă/recursivitate, arborele DFS, timpii de descoperire/finalizare. Componentele tare conexe (SCC): algoritmul Kosaraju (două DFS-uri), algoritmul Tarjan (un DFS cu stivă). Componentele biconexe: puncte de articulație, poduri.	2	Prelegere interactivă	
Tema 7: Drumuri minime. Algoritmul Dijkstra: cu heap binar $O((V+E) \log V)$, cu heap Fibonacci $O(E + V \log V)$. Proba de corectitudine prin invariantul de buclă. Algoritmul Bellman-Ford: relaxarea tuturor muchiilor de $V-1$ ori, detectarea ciclurilor negative, $O(VE)$. Algoritmul Floyd-Warshall: DP pe toate perechile, $O(V^3)$, detectarea ciclurilor negative. Compararea algoritmilor: densitatea grafului, greutate negative.	2	Prelegere, demonstrații	
Tema 8: Arbori de acoperire minimă și fluxuri. MST: proprietatea ciclului și a tăieturii — demonstrarea că Kruskal și Prim sunt corecți. Fluxuri în rețele: rețeaua reziduală, calea de augmentare, teorema max-flow min-cut (Ford-Fulkerson). Algoritmul Edmonds-Karp (BFS): $O(VE^2)$. Aplicații ale fluxului maxim: cuplaj bipartit maxim, acoperire minimă cu muchii.	2	Prelegere interactivă	
Tema 9: Structuri de date avansate. Heap Fibonacci: operații de complexitate amortizată $O(1)$ pentru insert și decrease-key, $O(\log n)$ pentru extract-min. Arbori de interval (Segment Tree): construcție $O(n)$, query și update $O(\log n)$, lazy propagation. Arbori indexați binar (Fenwick/BIT): $O(\log n)$ per operație, simplitatea implementării. Arbori van Emde Boas: $O(\log \log U)$ pentru operații pe mulțimi de întregi.	2	Prelegere interactivă	
Tema 10: Tehnici avansate de DP. DP cu optimizare: Divide and Conquer Optimization (complexitate $O(n \log n)$ în loc de $O(n^2)$). Knuth-Yao Optimization pentru probleme cu condiția cvadrangulă. DP pe mulțimi (bitmask DP): problema comis-voiajorului	2	Prelegere interactivă	

(TSP) — $O(2^n \times n^2)$. DP cu profil (broken profile DP): numărarea tilingurilor de table de dimensiune $n \times m$. DP pe arbore cu rerooting.			
Tema 11: Algoritmi de șiruri. Funcția de eșec KMP (Knuth-Morris-Pratt): construcția $O(m)$, căutarea $O(n)$, analiza corectitudinii. Algoritmul Z: Z-array, aplicații. Algoritmul Rabin-Karp: hashing polinomial, coliziuni, complexitate medie $O(n+m)$. Suffix Array: construcția $O(n \log n)$ cu algoritm SA-IS sau DC3. Suffix Automaton: construcția $O(n)$, aplicații (numărul de subșiruri distincte, cel mai lung subșir comun).	2	Prelegere interactivă	
Tema 12: Algoritmi probabilistici și aproximativi. Algoritmi Las Vegas vs. Monte Carlo. Quick Sort randomizat: analiza complexității așteptate $O(n \log n)$. Algoritmul Karger pentru min-cut: $O(V^2E)$. Hashing universal: rezolvarea coliziunilor, tabele de hashing cu $O(1)$ așteptat. Algoritmi de aproximare: schema de aproximare pentru knapsack, algoritmul greedy pentru acoperire cu mulțimi (set cover) — raportul $\ln(n)$.	2	Prelegere interactivă	
Tema 13: Clase de complexitate și intractabilitate. Clasa P și NP: definiții formale. Reducerile polinomiale. Problemele NP-complete: teorema Cook-Levin (SAT este NP-complet). Probleme NP-complete clasice: 3-SAT, Independent Set, Vertex Cover, Clique, Hamiltonian Path, TSP. Clasa NP-hard. Clasa co-NP. Strategii pentru probleme NP-hard: backtracking, branch-and-bound, parametrizare, aproximare.	2	Prelegere interactivă	
Tema 14: Algoritmi paraleli și distribuți — recapitulare. Modelul PRAM (Parallel Random Access Machine): CREW, CRCW. Algoritmi paraleli: prefix sum $O(\log n)$ cu $n/2$ procesoare, merge sort paralel. Speed-up, eficiență, legea lui Amdahl. Map-Reduce: paradigma, aplicații (word count, graph processing). Algoritmi distribuți: consensul în prezența defecțiunilor (teorema Fischer-Lynch-Paterson). Recapitulare generală.	2	Prelegere, recapitulare	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> 1. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C., Introduction to Algorithms (CLRS), 4th ed., MIT Press, 2022. 2. Kleinberg, J., Tardos, É., Algorithm Design, Pearson, 2005. 3. Skiena, S.S., The Algorithm Design Manual, 3rd ed., Springer, 2020. <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Sedgewick, R., Wayne, K., Algorithms, 4th ed., Addison-Wesley, 2011. Disponibil parțial gratuit la: https://algs4.cs.princeton.edu/ 5. Dasgupta, S., Papadimitriou, C., Vazirani, U., Algorithms, McGraw-Hill, 2006. Disponibil gratuit la: https://people.eecs.berkeley.edu/~vazirani/algorithms/all.pdf <p>Resurse software:</p> <ol style="list-style-type: none"> 6. Visualgo — vizualizarea algoritmilor și structurilor de date: https://visualgo.net/ 7. CP-algorithms — referință de algoritmi pentru programare competitivă: https://cp-algorithms.com/ 8. LeetCode — platformă de probleme algoritmice cu testare automată: https://leetcode.com/ 		

6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Divide et Impera — implementare și analiză. Implementarea în C++ a Merge Sort și Quick Sort (cu pivot median-of-3). Măsurarea timpilor de execuție pe tablouri de dimensiuni 10^3 , 10^4 , 10^5 și 10^6 elemente (aleatorii, sortate, invers sortate). Verificarea experimentală a complexității $O(n \log n)$ prin trasarea graficelor timp vs. n. Compararea cu <code>std::sort</code> .	2	Lucrare practică C++	
Lucrarea L2: Programare dinamică — implementare și comparare. Implementarea problemei rucsacului 0/1 cu memoizare (top-down) și tabulare (bottom-up). Implementarea LCS cu reconstruirea soluției. Compararea timpilor de execuție top-down vs. bottom-up pe instanțe mari. Rezolvarea a 3 probleme DP de pe LeetCode (nivel medium).	2	Lucrare practică C++, LeetCode	
Lucrarea L3: Algoritmi pe grafuri — BFS, DFS, SCC. Implementarea BFS și DFS pe grafuri reprezentate cu liste de adiacență. Implementarea algoritmului Kosaraju pentru componentele tare conexe. Testarea pe grafuri generate aleatoriu și verificarea numărului de SCC. Rezolvarea a 2 probleme de grafuri de pe LeetCode (nivel medium).	2	Lucrare practică C++	
Lucrarea L4: Drumuri minime. Implementarea Dijkstra cu <code>priority_queue</code> (heap binar). Implementarea Bellman-Ford cu detectarea ciclurilor negative. Testarea pe grafuri cu și fără greutate negative: compararea rezultatelor și a timpilor. Rezolvarea problemei Single Source Shortest Path pe un graf ponderat dat.	2	Lucrare practică C++	
Lucrarea L5: Structuri de date avansate — Segment Tree și BIT. Implementarea unui Segment Tree pentru query de suma pe interval și update punct. Implementarea unui BIT (Fenwick Tree) pentru același scop. Compararea complexității și a constantelor ascunse: benchmarking pe 10^6 operații. Extinderea Segment Tree cu lazy propagation pentru update pe interval.	2	Lucrare practică C++	
Lucrarea L6: Algoritmi de șiruri. Implementarea KMP: construcția funcției de eșec și căutarea. Implementarea Rabin-Karp cu hashing dublu pentru reducerea coliziunilor. Compararea timpilor pe texte mari (1MB) cu pattern-uri de diferite lungimi. Testarea KMP pe cazul cel mai defavorabil față de algoritmul naiv.	2	Lucrare practică C++	
Lucrarea L7: Proiect integrat și colocviu. Implementarea unui algoritm complex la alegere: TSP cu bitmask DP, flux maxim Edmonds-Karp, sau Suffix Array cu construcție $O(n \log n)$. Analiza teoretică a complexității (derivarea completă),	2	Proiect individual, colocviu	

implementarea cu teste pe cazuri de frontieră și cazul cel mai defavorabil, documentarea deciziilor de implementare (alegerea structurilor de date, optimizările aplicate). Colocviu oral.			
	<p>Bibliografie obligatorie laborator</p> <p>1. Cormen, T.H. et al., Introduction to Algorithms (CLRS), 4th ed., MIT Press, 2022 — exercițiile practice.</p> <p>2. CP-algorithms — implementări de referință și explicații detaliate: https://cp-algorithms.com/</p> <p>3. LeetCode — probleme algoritmice cu testare automată: https://leetcode.com/</p> <p>Bibliografie complementara laborator</p> <p>4. Sedgewick, R., Wayne, K., Algorithms, 4th ed., Addison-Wesley, 2011 — codul sursă Java disponibil: https://algs4.cs.princeton.edu/</p> <p>5. Codeforces — platformă de programare competitivă cu arhivă de probleme: https://codeforces.com/</p> <p>Resurse online gratuite recomandate</p> <p>6. CP-algorithms — referință de algoritmi și implementări: https://cp-algorithms.com/</p> <p>7. Visualgo — vizualizarea interactivă a algoritmilor: https://visualgo.net/</p> <p>8. LeetCode — probleme algoritmice: https://leetcode.com/</p> <p>9. Codeforces — probleme de programare competitivă: https://codeforces.com/</p>		

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

Proiectarea Algoritmilor aprofundează competențele din Structuri de Date și Algoritmi cu tehnicile avansate de proiectare — programare dinamică, greedy, divide-et-impera, fluxuri, algoritmi de șiruri — și cu analiza riguroasă a complexității. Aceste tehnici sunt solicitate direct în interviurile tehnice pentru poziții de software engineer la companii de top și constituie fundamentul oricărui sistem care trebuie să funcționeze la scară.

Combinarea analizei teoretice riguroase (teorema master, proba de corectitudine greedy, demonstrarea NP-completitudinii) cu implementarea și benchmarking-ul experimental în laborator formează capacitatea de a evalua critic eficiența soluțiilor propuse — direct corelată cu A6 și C8 din matricea de corelare.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea și aplicarea corectă a teoremei master (toate cazurile) și a metodei arborelui de recursivitate; proiectarea și analiza algoritmilor DP (identificarea subproblemelor, formularea recurenței, analiza complexității); cunoașterea și aplicarea algoritmilor greedy cu proba de corectitudine; cunoașterea algoritmilor de grafuri (Dijkstra, Bellman-Ford, Floyd-Warshall, Kruskal, Kosaraju); cunoașterea	Examen scris (2 ore): 1 problemă de analiză a complexității (aplicarea teoremei master sau rezolvarea recurenței prin arborele de recursivitate), 1 problemă de programare dinamică (formularea recurenței, tabela DP și analiza complexității pentru o problemă nouă) și 1 problemă de grafuri sau greedy (proiectarea algoritmului cu justificarea corectitudinii)	60%

	claselor P, NP și a reducerilor polinomiale.		
8.5 Seminar/laborator	Corectitudinea implementărilor C++ (codul compilează și produce rezultatele corecte pentru toate cazurile de test, inclusiv cazurile de frontieră); calitatea analizei experimentale (grafice timp vs. n cu interpretarea corectă față de complexitatea teoretică); calitatea proiectului integrat din L7 (derivarea completă a complexității și documentarea deciziilor de implementare); calitatea prezentării la colocviu.	Evaluare continuă: implementări individuale predate pe Sakai după fiecare lucrare (50%) și proiect integrat cu colocviu oral individual la L7 (50%)	40%
8.6 Standard minim de performanță:			
Predarea la colocviul din L7 a derivării complete a complexității algoritmului ales — nu doar enunțarea rezultatului, ci demonstrarea pas cu pas (recurență, aplicarea teoremei master sau analiza directă) — ȘI identificarea unui caz de frontieră netrivial cu comportament diferit față de cazul obișnuit. La examenul scris: formularea recurenței DP pentru problema dată cu explicarea semnificației fizice a fiecărei variabile de stare, chiar dacă calculul tabelii este incomplet.			

Data completării	Semnătura titularului de curs Lect. Dr. Panfiloiu Gheorghe	Semnătura titularului de seminar / laborator Lect. Dr. Panfiloiu Gheorghe
17.03.2026		
Data avizării în Consiliul Departamentului	Semnătura directorului DIT Lect. Dr. Antohe Valerian	
24.03.2026		
Data aprobării în Consiliul Facultății	Semnătura decanului FMDT Conf. Dr. Beteringhe Adrian	
07.04.2026		



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei	INGINERIA PROGRAMELOR – TIDS308						
1.2 Titularul activităților de curs	Conf. Dr. Lacrămă Dan Laurențiu						
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator	Conf. Dr. Lacrămă Dan Laurențiu						
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	C	1.8 Regimul disciplinei	Ob DS

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	3	din care: 2.2 curs	2	2.3 seminar/laborator	0/1
2.4 Total ore din planul de învățământ	42	din care: 2.5 curs	28	2.6 seminar/laborator	0/14
Distribuția fondului de timp					ore
2.7 Activități monitorizate					-
Practică					-
Elaborare proiect diplomă					-
2.8 Activități individuale					33
Documentare					10
Studiu individual					10
Referate					10
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					3
2.9 Total ore pe semestru					75
2.10 Numărul de credite					3

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Programare Orientată pe Obiecte (TIDS204) — clase, interfețe, design patterns de bază; Structuri de Date și Algoritmi (TIDS203) — complexitate, structuri de date; Baze de Date (TIDS305) — modelul relațional și SQL pentru componentele de persistență ale proiectului din laborator.
3.2 de competențe	Capacitatea de a proiecta și implementa o aplicație OOP de complexitate medie; cunoașterea a cel puțin 3 design patterns (Singleton, Observer, Factory); experiența lucrului cu Git la nivel de bază (clone, commit, push, pull, branch); familiarizarea cu cel puțin un IDE modern (IntelliJ, VS Code sau Eclipse).

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu tablă, videoproiector și acces la internet; platforma Sakai; recomandare: Git și un IDE modern instalate înainte de curs, cont GitHub sau GitLab activ.
4.2 de desfășurare a seminarului/laboratorului	Laborator de informatică cu calculatoare individuale; Git, Java/Python, IDE (IntelliJ IDEA sau VS Code), Maven sau Gradle instalate; studenții

	pot folosi propriul laptop; platforma GitHub sau GitLab pentru gestiunea codului sursă.
--	---

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C9. Posedă cunoștințe avansate de ingineria programelor — ciclul de viață al software-ului, metode agile, testare și asigurarea calității — cu conștientizarea critică a provocărilor actuale din industria software.
5.2 Aptitudini	A5. Utilizează biblioteci software, framework-uri și instrumente specifice pentru eficientizarea procesului de dezvoltare și adaptarea soluțiilor la cerințele tehnice și funcționale ale beneficiarilor. A25. Gestionează proiecte de inginerie software de complexitate medie și ridicată, planificând activitățile, alocând resursele, monitorizând progresul și aplicând metode agile pentru livrarea la termen a produselor software de calitate.
5.3 Responsabilitate și autonomie	R9. Gestionează autonom situații tehnice complexe și imprevizibile — defecțiuni critice de sistem, vulnerabilități de securitate, degradări de performanță — adoptând noi abordări strategice și luând decizii rapide și documentate.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Ingineria programelor — domeniu și provocări. Definiția ingineriei software (IEEE). Diferența față de programare: complexitate, lucru în echipă, întreținere. Crizele software: statistici actuale (CHAOS Report). Proprietățile calității software: funcționalitate, fiabilitate, utilizabilitate, eficiență, întreținabilitate, portabilitate (ISO/IEC 25010). Datoria tehnică (technical debt): conceptul și consecințele.	2	Prelegere interactivă	
Tema 2: Ciclul de viață al software-ului. Modelul cascadă: faze, avantaje și limitări. Modelul V: corelarea fazelor de dezvoltare cu testarea. Modelul spiral (Boehm): managementul riscului. Modele iterative: RUP (Rational Unified Process). Prototiparea. Compararea modelelor: criterii de alegere în funcție de tipul proiectului, mărimea echipei și gradul de incertitudine al cerințelor.	2	Prelegere interactivă	
Tema 3: Metode agile. Manifestul Agile: 4 valori și 12 principii. Scrum: roluri (Product Owner, Scrum Master, Echipa de dezvoltare), artefacte (Product Backlog, Sprint Backlog, Increment), ceremonii (Sprint Planning, Daily Scrum, Sprint Review, Retrospectivă), sprint-ul. Kanban: boardul vizual, limitarea WIP. Extreme Programming (XP): pair programming, TDD, continuous integration. SAFe (Scaled Agile Framework).	2	Prelegere interactivă	
Tema 4: Ingineria cerințelor. Tipuri de cerințe: funcționale, non-funcționale, de constrângere. Tehnici de elicitare: interviuri, chestionare, analiza documentelor, prototipare, workshop-	2	Prelegere interactivă	

uri. User stories: format (As a..., I want..., So that...), criterii de acceptanță, story points. Use case diagrams (UML): actori, cazuri de utilizare, relații (include, extend). Specificația cerințelor software (SRS): structura IEEE 830.			
Tema 5: Proiectarea arhitecturală. Conceptul de arhitectură software: componente, conectori, constrângeri. Atributele de calitate arhitecturală: performanță, disponibilitate, securitate, modificabilitate. Stiluri arhitecturale: layered (N-tier), client-server, microservicii, event-driven, pipes-and-filters. Architecture Decision Records (ADR). Documentarea arhitecturii: diagramele C4 (Context, Container, Component, Code).	2	Prelegere interactivă	
Tema 6: Design patterns — creaționale și structurale. Conceptul de design pattern (GoF). Patterns creaționale: Singleton (probleme cu testabilitatea), Factory Method, Abstract Factory, Builder, Prototype. Patterns structurale: Adapter, Bridge, Composite, Decorator, Facade, Flyweight, Proxy. Analiza comparativă: când se aplică fiecare, forțe, consecințe. Anti-patterns: God Object, Spaghetti Code, Golden Hammer.	2	Prelegere interactivă	
Tema 7: Design patterns — comportamentale și principii SOLID. Patterns comportamentale: Observer, Strategy, Command, Template Method, Iterator, State, Chain of Responsibility, Visitor. Principiile SOLID: Single Responsibility, Open/Closed, Liskov Substitution, Interface Segregation, Dependency Inversion — cu exemple de cod. Legea lui Demeter. Coeziune și cuplare: definiții, măsurarea, îmbunătățirea.	2	Prelegere interactivă	
Tema 8: Calitatea codului și refactoring. Cod curat (Clean Code): numele sugestive, funcțiile scurte, absența comentariilor inutile. Code smells: Long Method, Large Class, Feature Envy, Duplicate Code, Data Clumps, Primitive Obsession. Tehnici de refactoring (Fowler): Extract Method, Extract Class, Move Method, Replace Conditional with Polymorphism, Introduce Parameter Object. Refactoring cu teste de regresie.	2	Prelegere interactivă	
Tema 9: Testarea software. Tipuri de testare: unitară, de integrare, de sistem, de acceptanță. Testarea unității cu JUnit 5 sau pytest: adnotări, assert-uri, fixtures. Test-Driven Development (TDD): ciclul Red-Green-Refactor, beneficii și limitări. Acoperirea codului (code coverage): line, branch, mutation testing. Testarea bazată pe specificație: partiționarea echivalenței, analiza valorilor de frontieră.	2	Prelegere interactivă	

<p>Tema 10: Integrare și livrare continuă (CI/CD). Controlul versiunilor: Git avansat — branching strategies (Git Flow, trunk-based development), pull requests, code review. Continuous Integration: conceptul, serverele CI (GitHub Actions, Jenkins, GitLab CI). Continuous Delivery vs. Continuous Deployment. Pipeline CI/CD: build, test, static analysis (SonarQube), containerizare (Docker), deployment. Rollback și feature flags.</p>	2	Prelegere interactivă	
<p>Tema 11: Managementul proiectelor software. Estimarea efortului: story points, Planning Poker, metoda COCOMO II, analogia. Planificarea iterativă: release planning, sprint planning. Gestionarea riscurilor: identificare, evaluare (probabilitate × impact), strategii (evitare, mitigare, transfer, acceptare). Monitorizarea progresului: velocity, burndown chart, cumulative flow diagram. Gestionarea echipelor: formarea, storming, norming, performing (Tuckman).</p>	2	Prelegere interactivă	
<p>Tema 12: Documentarea software. Tipuri de documentație: pentru utilizatori (ghid de utilizare, FAQ), pentru dezvoltatori (API docs, arhitecturală), pentru operare (runbooks). JavaDoc și docstrings: standardele de documentare a codului. Documentarea API REST: OpenAPI/Swagger. Documentele de arhitectură: Architecture Decision Records, diagrame C4. Datoria de documentare: costul lipsei documentației.</p>	2	Prelegere interactivă	
<p>Tema 13: Metrici și managementul calității software. Metrici de produs: linii de cod (LOC), complexitate ciclomatică (McCabe), Halstead, CK metrics (LCOM, CBO, RFC, DIT, NOC). Metrici de proces: densitatea defectelor, viteza de rezolvare, velocity. SonarQube: calculul technical debt, code smells, vulnerabilități, duplicări. Total Quality Management (TQM) în software. ISO/IEC 25010 și modelul de calitate.</p>	2	Prelegere interactivă	
<p>Tema 14: Tendințe și recapitulare. DevOps: cultura, principiile (CALMS), toolchain-ul. Site Reliability Engineering (SRE): SLI, SLO, SLA, error budgets. Ingineria software pentru AI: provocări specifice (data versioning, model testing, ML pipelines). Low-code și no-code: implicații pentru ingineria software. Viitorul profesiei de software engineer în era AI. Recapitulare generală.</p>	2	Prelegere, recapitulare	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> 1. Sommerville, I., Software Engineering, 10th ed., Pearson, 2015. 2. Martin, R.C., Clean Code: A Handbook of Agile Software Craftsmanship, Prentice Hall, 2008. 3. Gamma, E., Helm, R., Johnson, R., Vlissides, J., Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1994. 		

	<p>Bibliografie complementară</p> <p>4. Fowler, M., Refactoring: Improving the Design of Existing Code, 2nd ed., Addison-Wesley, 2018.</p> <p>5. Beck, K., Test-Driven Development: By Example, Addison-Wesley, 2002.</p> <p>6. Kim, G., Humble, J., Behr, P., Forsgren, N., The DevOps Handbook, 2nd ed., IT Revolution Press, 2021.</p> <p>Resurse software:</p> <p>7. GitHub Documentation — Git, GitHub Actions, pull requests: https://docs.github.com/</p> <p>8. SonarQube Documentation — analiza statică a codului: https://docs.sonarqube.org/</p> <p>9. Martin Fowler — articole despre refactoring, arhitectură și patterns: https://martinfowler.com/</p>		
6.2 Laborator	Nr. ore	Metode de predare	Observații
<p>Lucrarea L1: Git avansat și workflow colaborativ. Configurarea unui repository GitHub cu branching strategy Git Flow. Crearea de branch-uri feature/, release/, hotfix/. Simularea unui conflict de merge și rezolvarea lui. Configurarea unui pull request cu code review: comentarii linie-cu-linie, cerința de aprobare. Configurarea unui fișier .gitignore și a unui GitHub Actions workflow simplu (build + test).</p>	2	Lucrare practică Git, GitHub	
<p>Lucrarea L2: Design patterns — implementare. Implementarea în Java sau Python a 4 design patterns pe o temă de aplicație comună (ex: sistem de notificări): Observer (pentru notificări), Strategy (pentru canale de trimitere), Factory (pentru tipuri de notificări), Decorator (pentru formatarea mesajelor). Justificarea alegerii fiecărui pattern față de alternativa fără pattern.</p>	2	Lucrare practică Java/Python	
<p>Lucrarea L3: Refactoring cu teste de regresie. Pornind de la un modul de cod cu 5 code smells identificabile (furnizat), scrierea mai întâi a testelor unitare cu JUnit 5 sau pytest care acoperă funcționalitatea existentă. Aplicarea refactoring-ului pas cu pas (Extract Method, Replace Conditional with Polymorphism). Verificarea că testele trec după fiecare pas. Compararea complexității ciclomatice înainte și după.</p>	2	Lucrare practică Java/Python	
<p>Lucrarea L4: TDD — Test-Driven Development. Implementarea unui modul (ex: parser de expresii aritmetice sau validator de date) prin ciclul TDD strict: Red (test eșuat) → Green (implementare minimă) → Refactor. Minim 10 teste scrise înainte de implementare. Analiza acoperirii codului cu JaCoCo (Java) sau coverage.py (Python): atingerea unui nivel de 85%.</p>	2	Lucrare practică TDD	
<p>Lucrarea L5: Pipeline CI/CD cu GitHub Actions. Configurarea unui pipeline complet în GitHub Actions pentru un proiect Java/Python: etapele build, unit test, coverage report, static analysis (SonarCloud sau Checkstyle), build Docker image. Interpretarea raportului SonarCloud:</p>	2	Lucrare practică GitHub Actions, Docker	

identificarea și corectarea a cel puțin 3 code smells și 1 vulnerabilitate de securitate detectate automat.			
Lucrarea L6: Estimare și planificare agile. Simularea unui sprint în echipă de 2-3 studenți: definirea unui Product Backlog de 10 user stories cu criterii de acceptanță, estimarea cu Planning Poker (story points), sprint planning pentru un sprint de o săptămână, burndown chart la finalul sprint-ului simulat. Documentarea: Definition of Done, retrospectivă scrisă.	2	Atelier agile, Jira/Trello	
Lucrarea L7: Proiect integrat și colocviu. Prezentarea unui proiect individual care aplică toate competențele din laborator: repository Git cu commit history clar și pull request codat review-uit, cel puțin 3 design patterns implementate și justificate, suită de teste unitare cu acoperire de minim 80%, pipeline CI/CD funcțional. Colocviu oral: justificarea deciziilor de proiectare, demonstrarea live a pipeline-ului.	2	Proiect individual, colocviu	
<p>Bibliografie obligatorie laborator</p> <ol style="list-style-type: none"> 1. Martin, R.C., Clean Code: A Handbook of Agile Software Craftsmanship, Prentice Hall, 2008 — capitolele de refactoring. 2. Fowler, M., Refactoring: Improving the Design of Existing Code, 2nd ed., Addison-Wesley, 2018. 3. GitHub Documentation — Actions, pull requests, code review: https://docs.github.com/ <p>Bibliografie complementara laborator</p> <ol style="list-style-type: none"> 4. Beck, K., Test-Driven Development: By Example, Addison-Wesley, 2002. 5. SonarQube/SonarCloud Documentation — analiza statică: https://docs.sonarqube.org/ <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. GitHub Documentation — Git, GitHub Actions: https://docs.github.com/ 7. Martin Fowler — refactoring catalog, patterns, arhitectură: https://martinfowler.com/ 8. JUnit 5 Documentation: https://junit.org/junit5/docs/current/user-guide/ 9. SonarCloud — analiza statică gratuită pentru proiecte open source: https://sonarcloud.io/ 			

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

<p>Ingineria Programelor acoperă competențele transversale care transformă un programator individual într-un inginer software capabil să lucreze în echipă pe produse de complexitate industrială: metode agile, design patterns, calitatea codului, testarea automată și CI/CD. Aceste competențe sunt solicitate explicit în orice interviu de software engineer și constituie fundamentul colaborării eficiente în echipele de dezvoltare.</p> <p>Componenta de laborator, centrată pe un proiect individual cu repository Git, design patterns justificate, TDD și pipeline CI/CD, urmărește formarea obiceiurilor de lucru profesionale — direct corelate cu A5, A25 și R9 din matricea de corelare — nu doar cunoașterea lor teoretică.</p>

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea și compararea modelelor de ciclu de viață	Colocviu oral (20 minute): justificarea orală a deciziilor de proiectare din proiectul de	40%

	(cascadă, agile, spiral); cunoașterea ceremoniilor și rolurilor Scrum; cunoașterea și aplicarea design patterns GoF (creaționale, structurale, comportamentale) cu justificarea alegerii; cunoașterea principiilor SOLID; cunoașterea tehnicilor de testare (TDD, acoperire, tipuri de testare); cunoașterea conceptelor CI/CD și a instrumentelor DevOps.	laborator — alegerea design patterns, structura testelor, configurarea pipeline-ului — cu o întrebare suplimentară de identificare a unui code smell și propunerea de refactoring	
8.5 Seminar/laborator	Calitatea repository-ului Git (commit history clar, cel puțin un pull request cu comentarii de code review, branching strategy respectată); corectitudinea și justificarea design patterns (cel puțin 3 patterns cu explicarea forțelor și consecințelor fiecăruia); acoperirea codului de minim 80% cu teste unitare semnificative; pipeline CI/CD funcțional cu cel puțin 3 etape (build, test, static analysis).	Evaluare continuă: livrabile individuale predate pe Sakai după fiecare lucrare (50%) și proiect integrat cu colocviu oral individual la L7 (50%)	60%
8.6 Standard minim de performanță:			
Predarea unui repository Git public cu un pipeline CI/CD funcțional care execută automat testele unitare la fiecare push — pipeline-ul trebuie să aibă cel puțin etapele build și test și să afișeze statusul verde pe branch-ul main. Proiectele fără pipeline CI/CD configurabil sau cu zero teste unitare nu îndeplinesc standardul minim, indiferent de calitatea codului sursă. Această cerință diferă de standardele din alte discipline prin focusul pe uneltele de automatizare, nu pe conținutul algoritmic.			

Data completării 16.03.2026	Semnătura titularului de curs Conf. Dr. Lacrămă Dan Laurențiu	Semnătura titularului de seminar / laborator Conf. Dr. Lacrămă Dan Laurențiu
Data avizării în Consiliul Departamentului 24.03.2026	Semnătura directorului DIT Lect. Dr. Antohe Valerian	
Data aprobării în Consiliul Facultății 07.04.2026	Semnătura decanului FMDT Conf. Dr. Beteringhe Adrian	



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei				PRACTICĂ II– TIDS317			
1.2 Titularul activităților de curs							
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator / proiect / practică				Lect. Dr. Rusescu Ciprian			
1.5 Anul de studiu	II	1.6 Semestrul	1	1.7 Tipul de evaluare	C	1.8 Regimul disciplinei	Ob DS

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână		din care: 2.2 curs		2.3 seminar/laborator/proiect	
2.4 Total ore din planul de învățământ	60 ore	din care: 2.5 curs		2.6 seminar/laborator/proiect./ practică	60 ore
Distribuția fondului de timp					ore
2.7 Activități monitorizate					60
Practică					60
Elaborare proiect diplomă					
2.8 Activități individuale					15
Documentare					5
Studiu individual					
Referate					
Teme casă					
Proiect					8
Alte activități (inclusive colocviu, examen)					2
2.9 Total ore pe semestru					75
2.10 Numărul de credite					3

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Practică I (TIDS213) — primul contact cu mediul profesional; Programare Orientata pe Obiecte (TIDS204); Baze de Date (TIDS305); Ingineria Programelor (TIDS308); Rețele de Calculatoare (TIDS301).
3.2 de competențe	Capacitatea de a scrie cod în cel puțin un limbaj de programare la nivel mediu-avansat; cunoașterea fluxului de lucru Git (branch, pull request, code review); înțelegerea ciclului de viață software și a metodelor agile; abilitatea de a redacta documentație tehnică structurată.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	
4.2 de desfășurare a seminarului/laboratorului	Practica se desfășoară la organizația gazdă (companie parteneră sau instituție publică), conform convenției de practică semnate, în ritm de 4-5 ore pe săptămână, pe parcursul celor 14 săptămâni ale semestrului (60 ore total). Studentul lucrează sub îndrumarea unui tutore din organizație și a

	coordonatorului de practică din universitate. Activitățile pot fi desfășurate la sediul organizației sau parțial în regim remote, cu acordul tutorelui. Platforma Sakai este folosită pentru predarea documentelor obligatorii.
--	---

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C25. Posedă cunoștințe integrate și critice asupra domeniului tehnologiei informației ca întreg, cu capacitatea de a identifica conexiunile dintre subdomeniile disciplinei, de a evalua soluții tehnice alternative și de a formula direcții de cercetare sau dezvoltare originale, la avangarda domeniului.
5.2 Aptitudini	A25. Gestionează proiecte de inginerie software de complexitate medie și ridicată, planificând activitățile, alocând resursele, monitorizând progresul și aplicând metode agile pentru livrarea la termen a produselor software de calitate.
5.3 Responsabilitate și autonomie	R2. Gestionează independent ciclul complet de dezvoltare al unui produs software — de la analiza cerințelor până la testare și livrare — demonstrând capacitatea de a anticipa riscurile tehnice și de a propune soluții alternative viabile. R25. Își asumă în mod conștient și responsabil rolul de profesionist în domeniul tehnologiei informației ca agent al transformării digitale a societății, contribuind la dezvoltarea durabilă a comunității prin soluții informatice inovatoare, sigure și etice.

6. Conținuturi

6.1 Practică	Nr. ore	Metode de predare	Observații
Săptămânile 1-2: Integrare în organizație. Prezentarea organizației: structura, domeniul de activitate, produsele/serviciile, echipele tehnice. Cunoașterea politicilor interne: regulamente, securitatea informației, confidențialitatea, uneltele utilizate. Configurarea mediului de lucru: conturi, acces la repository, VPN, IDE, pipeline CI/CD. Identificarea proiectului sau a sarcinilor atribuite pentru întreagă perioadă de practică.	8	Activitate la organizația gazdă	
Săptămânile 3-5: Familiarizarea cu proiectul și tehnologiile. Studiarea documentației tehnice a proiectului: arhitectura, codul existent, convențiile de cod. Rularea și depanarea mediului local de dezvoltare. Identificarea și rezolvarea primelor sarcini de complexitate redusă (bug-uri, îmbunătățiri minore). Participarea la ceremoniile agile ale echipei: daily standup, sprint planning, retrospectiva.	12	Activitate la organizația gazdă	
Săptămânile 6-8: Implementarea sarcinilor atribuite — faza 1. Implementarea independentă a sarcinilor alocate: dezvoltare de funcționalități noi, refactorizare, scriere de teste unitare sau de integrare. Deschiderea de pull request-uri și parcurgerea procesului de code review. Documentarea progresului în caietul de practică. Discuții săptămânale cu tutorii pentru feedback și ajustarea direcției.	12	Activitate la organizația gazdă	
Săptămânile 9-11: Implementarea sarcinilor atribuite — faza 2. Continuarea și finalizarea	12	Activitate la organizația gazdă	

sarcinilor din sprint-ul curent. Tratarea feedback-ului din code review: revizuire, optimizari, corectarea erorilor identificate. Intelegerea si aplicarea standardelor de calitate ale organizației: coding style, acoperire cu teste, criterii de acceptanță. Participarea la demo-ul de sprint si prezența contribuțiilor.			
Săptămânile 12-13: Contribuții avansate și documentare. Asumarea unor sarcini de complexitate mai ridicată: proiectarea unei funcționalități mici, integrarea cu un serviciu extern, optimizarea performanței. Documentarea tehnică a contribuțiilor: comentarii în cod, actualizarea documentației interne (README, wiki). Redactarea secțiunilor principale ale raportului de practică.	8	Activitate la organizația gazdă	
Săptămâna 14: Finalizare si evaluare. Finalizarea tuturor sarcinilor asumate si predarea lor în starea agreata cu tutorele. Completarea caietului de practică: activitățile desfășurate săptămânal, competențele dobândite, reflecția personală asupra experienței. Obținerea evaluării si a semnăturii tutorelui din organizație. Predarea raportului final și a caietului de practică pe platforma Sakai.	8	Activitate la organizația gazdă	
	<p>Bibliografie obligatorie practică</p> <ol style="list-style-type: none"> 1. Regulamentul de practică al Universitatii Internationale Danubius — disponibil pe platforma Sakai. 2. Conventia de practică individuala — model disponibil la secretariatul facultatii. 3. Ghidul de redactare a caietului și a raportului de practică — disponibil pe platforma Sakai. <p>Bibliografie complementara practică</p> <ol style="list-style-type: none"> 4. Sommerville, I., Software Engineering, 10th ed., Pearson, 2015 — capitolele privind managementul proiectelor si lucrul în echipă. 5. Kim, G., Humble, J., Behr, P., Forsgren, N., The DevOps Handbook, 2nd ed., IT Revolution Press, 2021. <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. GitHub Documentation — ghid complet Git și colaborare: https://docs.github.com/ 7. Atlassian Git Tutorials — fluxuri de lucru în echipă: https://www.atlassian.com/git/tutorials 8. Scrum Guide (Ken Schwaber, Jeff Sutherland) — ghid oficial Scrum gratuit: https://scrumguides.org/ 		

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

Practică II plaseaza studentul din An III într-un mediu profesional real, cu responsabilitati concrete într-o echipă de dezvoltare software. Cele 60 de ore distribuite pe parcursul semestrului reproduc ritmul de lucru al unui junior developer part-time, permițând acumularea progresiva a experienței profesionale fara întreruperea cursurilor.

Participarea la ceremoniile agile, parcurgerea procesului de code review si asumarea unor sarcini de complexitate crescanda formeaza direct competențele A25 (gestiunea proiectelor software) si R2 (gestionarea independenta a ciclului de dezvoltare). Reflectia din caietul de practică si angajamentul fata de standardele organizației formeaza R25.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs			
8.5 Proiect / practică	Prezenta și activitatea la organizația gazdă (60 ore confirmate în caietul de practică cu semnăturile tutorelui, săptămână cu săptămână); calitatea contribuțiilor tehnice (cel puțin un pull request acceptat sau o sarcină finalizată și confirmată de tutore); calitatea și completitudinea caietului de practică (activitățile săptămânale documentate, competențele identificate, reflecția personală); calitatea raportului final de practică.	Evaluare pe baza: caietului de practică completat și semnat de tutore (40%) + raportul final de practică predat pe Sakai (30%) + evaluarea tutorelui din organizație (30%). Nota finală se acordă după confirmarea celor 60 de ore.	100%

8.6 Standard minim de performanță:

Confirmarea celor 60 de ore de practică prin caietul de practică completat săptămânal și semnat de tutorele din organizație, însoțit de cel puțin o contribuție tehnică concretă documentată (sarcină finalizată, cod livrat, funcționalitate testată). Un caiet de practică fără semnătura tutorelui sau fără descrierea activităților tehnice desfășurate nu îndeplinește standardul minim, indiferent de calitatea raportului final.

Data completării	Semnătura titularului de curs	Semnătura titularului de seminar / laborator/proiect/practică Lect. Dr. Rusescu Ciprian
19.03.2026		
Data avizării în Consiliul Departamentului	Semnătura directorului DIT Lect. dr. Antohe Valerian	
24.03.2026		
Data aprobării în Consiliul Facultății	Semnătura decanului FMDT Conf. dr. Beteringhe Adrian	
07.04.2026		



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei		MICROSISTEME ELECTRONICE ȘI MECANICE – TIDS319					
1.2 Titularul activităților de curs		Conf. Dr. Lacrămă Dan Laurențiu					
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator		Conf. Dr. Lacrămă Dan Laurențiu					
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	C	1.8 Regimul disciplinei	Op DS

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	3	din care: 2.2 curs	2	2.3 seminar/laborator	0/1
2.4 Total ore din planul de învățământ	42	din care: 2.5 curs	28	2.6 seminar/laborator	0/14
Distribuția fondului de timp					ore
2.7 Activități monitorizate					-
Practică					-
Elaborare proiect diplomă					-
2.8 Activități individuale					8
Documentare					2
Studiu individual					2
Referate					2
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					2
2.9 Total ore pe semestru					50
2.10 Numărul de credite					2

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Teoria Sistemelor (TIDS302) — funcții de transfer, stabilitate, MATLAB/Simulink; Achiziția și Prelucrarea Datelor (TIDS304) — senzori, ADC/DAC, interfețe; Proiectarea cu Microprocesoare (TIDS215) — arhitecturi embedded, GPIO, SPI, I2C, PWM.
3.2 de competențe	Cunoașterea noțiunilor de bază din mecanică (forță, cuplu, inerție); înțelegerea funcționării circuitelor electronice analogice (amplificatoare operaționale, filtre) și digitale (logică combinațională și secvențială); abilitatea de a programa un microcontroler (Arduino sau STM32) și de a utiliza MATLAB.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu tablă, videoproiector și acces la internet; platforma Sakai; recomandare: MATLAB cu Simulink și Control System Toolbox instalate înainte de curs.
-------------------------------	---

4.2 de desfășurare a seminarului/laboratorului	Laborator dotat cu: plăci de dezvoltare Arduino Mega/STM32 Nucleo; motoare DC, servomotoare și motoare pas cu pas cu drivere (L298N, A4988); senzori (encoder incremental, IMU MPU-6050, senzor de forță); osciloscop digital; calculatoare cu MATLAB/Simulink și Arduino IDE.
--	--

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C29. Deține cunoștințe specializate privind teoria sistemelor — sisteme liniare, funcții de transfer, stabilitate, răspuns în frecvență — și aplicarea acestora în modelarea și analiza sistemelor de control și a microsistemelor. C31. Posedă cunoștințe de bază în mecanică și teoria mecanismelor, suficiente pentru înțelegerea principiilor funcționării sistemelor mecatronice, robotice și a componentelor de acționare din sistemele embedded.
5.2 Aptitudini	A29. Modelează și analizează sisteme dinamice utilizând tehnici din teoria sistemelor (funcții de transfer, diagrame Bode, spații de stare), utilizând instrumente software specializate (MATLAB/Simulink) pentru simularea comportamentului sistemelor de control.
5.3 Responsabilitate și autonomie	R29. Își asumă responsabilitatea pentru modelarea corectă a sistemelor dinamice în proiectele ingineresti, validând experimental modelele teoretice și documentând transparent discrepanțele dintre comportamentul simulat și cel real.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Introducere în microsisteme și mecatronică. Definiția microsistemului și a sistemului mecatronic. Componentele unui sistem mecatronic: mecanic, electric, electronic, informatic. Clasificarea acționărilor: electrice, hidraulice, pneumatice. Lanțul cinematic senzor–controller–actuator. Exemple industriale: roboți, drone, imprimante 3D.	2	Prelegere interactivă	
Tema 2: Elemente de mecanică pentru microsisteme. Cinematica și dinamica mișcării de translație și rotație. Momentul de inerție: calcul pentru forme geometrice uzuale. Cuplul și puterea mecanică. Transmisii mecanice: angrenaje (raportul de transmitere, eficiența), curele, șurub-piuliță (ball screw). Reducerea la axul motorului.	2	Prelegere interactivă	
Tema 3: Motoare de curent continuu (DC). Principiul de funcționare: legea lui Lorentz, ecuațiile electrice și mecanice. Caracteristica mecanică: viteza în gol, cuplul de blocare. Modelul de transfer al motorului DC: funcția de transfer tensiune→viteză. Parametrii motorului: constanta de cuplu K_m , constanta FEM K_e , rezistența de armătură R_a , inductanța L_a , inerția J .	2	Prelegere interactivă	
Tema 4: Controlul vitezei motoarelor DC. Modulația în lățime de impuls (PWM): principiu, frecvența de comutare, rezoluția. Driver-ul H-bridge (L298N): funcționare, protecția la supracurent. Controlul PID al	2	Prelegere interactivă	

vitezei: implementarea pe microcontroler. Encoder incremental: rezoluție, calculul vitezei prin numărare de impulsuri pe interval de timp.			
Tema 5: Servomotoare și controlul poziției. Servomotorul RC: structura (motor DC + reductor + potențiomtru + circuit de control), semnalul PWM de comandă (1-2 ms). Servomotorul industrial cu feedback encoder. Controlul poziției cu PID: structura, sintonizarea. Anti-windup. Profilul de mișcare trapezoidal: accelerare, mers uniform, decelerare.	2	Prelegere interactivă	
Tema 6: Motoare pas cu pas. Principiul de funcționare: stator cu bobine, rotor dințat. Tipuri: unipolare și bipolare. Modurile de pas: full step, half step, microstepping. Caracteristica viteză-cuplu: zona de start-stop și zona de funcționare. Driverul A4988/DRV8825: configurarea microstepping-ului, curentul de referință. Calculul vitezei și accelerației în pași pe secundă.	2	Prelegere interactivă	
Tema 7: Senzori de mișcare și forță. Encodere: incrementale (canal A, B, Z) și absolute (Gray code). Rezolvere și senzori de efect Hall. Tahogeneratoare. Senzori inițiali și de limită: microswitch-uri, senzori optici de barieră, senzori magnetici (Hall). Traductoare de forță și cuplu: senzori cu gauge strain, puntea Wheatstone, amplificatorul de instrumentare.	2	Prelegere interactivă	
Tema 8: Sisteme MEMS. Definiția MEMS (Micro-Electro-Mechanical Systems). Procese de fabricație: litografie, depunere, gravare. Accelerometre MEMS: principiul capacitiv, zgomotul de cuantizare, banda. Giroscop MEMS: efectul Coriolis, deriva (gyro drift). IMU (Inertial Measurement Unit): combinarea accelerometrului și giroscopului. Filtrarea complementară și filtrul Kalman pentru fuziunea senzorilor.	2	Prelegere interactivă	
Tema 9: Acționări hidraulice și pneumatice. Principiul hidraulic: legea lui Pascal, debitul, presiunea. Pompe hidraulice, cilindri hidraulici, supape de control direcțional. Servovalva hidraulică: caracteristica presiune-debit. Sisteme pneumatice: compresoare, regulatoare de presiune, cilindri pneumatici. Comparatie hidraulic-pneumatic-electric: putere specifică, răspuns, precizie.	2	Prelegere interactivă	
Tema 10: Controlul în buclă închisă al sistemelor mecatronice. Structura de control în cascadă: bucla de curent, bucla de viteză, bucla de poziție. Sinteza regulatorului PID în domeniul frecvență: marginea de câștig și de fază. Calculul funcției de transfer a motorului	2	Prelegere, MATLAB/Simulink	

DC cu reductor și sarcină. Simularea buclei de reglare în MATLAB/Simulink.			
Tema 11: Interfețe de comunicație pentru sisteme embedded. CAN bus: arbitrarea priorităților, structura cadrului, viteza. RS-485: magistrală multidrop, comunicația în rețele industriale. Modbus RTU: structura cererii și răspunsului, adresarea, registrele. EtherCAT: comunicație deterministă în timp real. Comparatie CAN–RS485–EtherCAT pentru aplicații de control al mișcării.	2	Prelegere interactivă	
Tema 12: Sisteme de poziționare și robotică. Cinematica unui robot cu 2 grade de libertate (2-DOF): cinematica directă și inversă (calcul analitic). Spațiul articular vs. spațiul cartezian. Generarea traiectoriei: interpolare liniară și circulară. Sisteme Cartesiene (CNC): structura mecanică, rezoluția, repetabilitatea. Delta robot: principiul de funcționare.	2	Prelegere interactivă	
Tema 13: Microsisteme pentru IoT și wearables. Procesoare de consum ultra-scăzut: ARM Cortex-M0+ (Nordic nRF52840). Protocoale wireless pentru microsisteme: Bluetooth Low Energy (BLE), Zigbee, LoRa. Managementul energiei: moduri sleep, harvesting de energie. Senzori wearable: PPG (puls), ECG, temperatură cutanată. Platforma Arduino Nano 33 IoT și STM32L4.	2	Prelegere interactivă	
Tema 14: Tendințe și recapitulare. Fabricația aditivă și microsistemele: imprimarea 3D a componentelor mecanice cu toleranțe strânse. Soft robotics: materiale elastomere, acționări pneumatice moi. Exoschelete și robotica medicală: cerințele de siguranță. Mașinile autonome: integrarea senzorilor MEMS, LIDAR, cameră. Recapitulare generală.	2	Prelegere, recapitulare	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> Ogata, K., Modern Control Engineering, 5th ed., Pearson, 2010. Craig, J.J., Introduction to Robotics: Mechanics and Control, 3rd ed., Pearson, 2004. Spong, M.W., Hutchinson, S., Vidyasagar, M., Robot Modeling and Control, 2nd ed., Wiley, 2020. <p>Bibliografie complementară</p> <ol style="list-style-type: none"> Nise, N.S., Control Systems Engineering, 8th ed., Wiley, 2019. Fraden, J., Handbook of Modern Sensors: Physics, Designs, and Applications, 5th ed., Springer, 2016. <p>Resurse software:</p> <ol style="list-style-type: none"> MathWorks Simulink Documentation — modelare și simulare sisteme dinamice: https://www.mathworks.com/help/simulink/ Arduino Documentation — referință completă hardware și software: https://docs.arduino.cc/ STM32 Documentation — microcontrolere STM32 și HAL: https://www.st.com/en/microcontrollers-microprocessors/stm32-32-bit-arm-cortex-mcus.html 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Control PWM al unui motor DC. Implementarea controlului PWM al unui motor DC cu driver L298N și Arduino. Varierea raportului de umplere și măsurarea vitezei cu encoder incremental. Trasarea	2	Lucrare practică Arduino	

curbei viteză–PWM. Identificarea experimentală a constantei de timp mecanice.			
Lucrarea L2: Regulator PID de viteză. Implementarea unui regulator PID de viteză pe Arduino pentru motorul DC din L1. Sintonizarea manuală a parametrilor K_p , K_i , K_d . Compararea răspunsului la treaptă cu și fără regulator. Analiza efectului fiecărui termen P, I, D.	2	Lucrare practică Arduino	
Lucrarea L3: Controlul poziției cu servomotor. Controlul poziției unui servomotor RC cu Arduino prin semnal PWM. Implementarea unui regulator PID de poziție cu encoder. Urmărirea unui profil de mișcare trapezoidal. Măsurarea erorii de poziție staționar.	2	Lucrare practică Arduino	
Lucrarea L4: Motor pas cu pas și microstepping. Controlul unui motor pas cu pas bipolar cu driver A4988 și Arduino: full step, half step, microstepping 1/16. Măsurarea vibrațiilor pentru fiecare mod. Implementarea unui profil de accelerare (ramping) pentru evitarea pierderii de pași.	2	Lucrare practică Arduino	
Lucrarea L5: Senzori inerțiali — IMU MPU-6050. Citirea datelor de la accelerometru și giroscop (MPU-6050) prin I2C cu Arduino. Calculul unghiurilor Euler din accelerometru. Observarea derivei giroscopului pe interval de timp. Implementarea filtrului complementar pentru fuziunea senzorilor.	2	Lucrare practică Arduino	
Lucrarea L6: Modelare și simulare în MATLAB/Simulink. Construirea modelului Simulink al unui motor DC cu sarcină: blocuri de transfer, integrare, feedback. Simularea răspunsului la treaptă cu și fără regulator PID. Compararea cu rezultatele experimentale din L2. Calculul marginilor de câștig și de fază din diagrama Bode.	2	Lucrare practică MATLAB/Simulink	
Lucrarea L7: Proiect integrat și colocviu. Implementarea unui sistem complet de poziționare cu motor DC, encoder și regulator PID: urmărire de traiectorie definită. Documentarea tehnică: schema bloc, funcția de transfer identificată experimental, curbele de performanță (răspuns la treaptă, eroarea de urmărire), discrepanțele față de modelul Simulink. Colocviu oral.	2	Proiect individual, colocviu	
	<p>Bibliografie obligatorie laborator</p> <p>1. Ogata, K., Modern Control Engineering, 5th ed., Pearson, 2010 — exercițiile practice.</p> <p>2. Arduino Documentation: https://docs.arduino.cc/</p> <p>3. MathWorks Simulink Documentation: https://www.mathworks.com/help/simulink/</p> <p>Bibliografie complementara laborator</p> <p>4. Fraden, J., Handbook of Modern Sensors, 5th ed., Springer, 2016.</p> <p>5. STM32 HAL Documentation: https://www.st.com/en/microcontrollers-microprocessors/stm32-32-bit-arm-cortex-mcus.html</p> <p>Resurse online gratuite recomandate</p> <p>6. Arduino Documentation: https://docs.arduino.cc/</p> <p>7. MathWorks Simulink: https://www.mathworks.com/help/simulink/</p>		

	8. STM32 Documentation: https://www.st.com/en/microcontrollers-microprocessors/stm32-32-bit-arm-cortex-mcus.html
	9. MPU-6050 Datasheet și ghid de utilizare — InvenSense: https://invensense.tdk.com/products/motion-tracking/6-axis/mpu-6050/

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

Microsistemele Electronice și Mecanice integrează teoria sistemelor cu componentele fizice reale ale acționărilor: motoare DC, servomotoare, motoare pas cu pas, senzori MEMS. Disciplina acoperă nișa dintre programarea embedded și ingineria mecanică, esențială pentru robotică, automatizare industrială și dispozitive IoT.

Componenta de laborator urmărește validarea experimentală a modelelor teoretice și documentarea transparentă a discrepanțelor față de simulare — direct corelată cu R29 din matricea de corelare.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea principiilor de funcționare ale motoarelor DC, pas cu pas și servomotoarelor; cunoașterea modelului de transfer al motorului DC și calculul funcției de transfer în buclă închisă cu regulator PID; cunoașterea principiului accelerometrelor și giroscopelor MEMS; cunoașterea transmisiilor mecanice și a reducerii la axul motorului; cunoașterea protocoalelor industriale CAN și Modbus.	Colocviu oral (20 minute): justificarea deciziilor de proiectare din lucrarea L7 — identificarea experimentală a modelului motorului, sintonizarea PID, discrepanța față de simularea Simulink — cu o întrebare suplimentară despre un scenariu de control al mișcării nou	40%
8.5 Seminar/laborator	Funcționalitatea sistemelor implementate în laborator (motorul se rotește la viteza sau poziția cerută, erorii de urmărire măsurate și raportate); calitatea documentației tehnice din L7 (schema bloc, funcția de transfer identificată, curbele de performanță, discrepanța față de Simulink explicată); calitatea prezentării la colocviu.	Evaluare continuă: rapoarte individuale predate pe Sakai după fiecare lucrare (50%) și proiect integrat cu colocviu oral individual la L7 (50%)	60%

8.6 Standard minim de performanță:

Demonstrarea la colocviul din L7 a sistemului de poziționare funcțional (motorul urmărește traiectoria definită cu eroare staționar nulă sau sub pragul specificat) și prezentarea comparației dintre răspunsul experimental și simularea Simulink, cu explicarea cauzei discrepantei. Un sistem care nu funcționează demonstrabil sau o documentație fără secțiunea de validare experimentală nu îndeplinesc standardul minim.

Data completării 17.03.2026	Semnătura titularului de curs Conf. Dr. Lacrămă Dan Laurențiu	Semnătura titularului de seminar / laborator Conf. Dr. Lacrămă Dan Laurențiu
Data avizării în Consiliul Departamentului 24.03.2026	Semnătura directorului DIT Lect. dr. Antohe Valerian	
Data aprobării în Consiliul Facultății 07.04.2026	Semnătura decanului FMDT Conf. dr. Beteringhe Adrian	



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei	SISTEME AUTOMATE – TIDS320						
1.2 Titularul activităților de curs	Conf. Dr. Lacrămă Dan Laurențiu						
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator	Conf. Dr. Lacrămă Dan Laurențiu						
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	C	1.8 Regimul disciplinei	Op DS

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	3	din care: 2.2 curs	2	2.3 seminar/laborator	0/1
2.4 Total ore din planul de învățământ	42	din care: 2.5 curs	28	2.6 seminar/laborator	0/14
Distribuția fondului de timp					ore
2.7 Activități monitorizate					-
Practică					-
Elaborare proiect diplomă					-
2.8 Activități individuale					8
Documentare					2
Studiu individual					2
Referate					2
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					2
2.9 Total ore pe semestru					50
2.10 Numărul de credite					2

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Teoria Sistemelor (TIDS302) — funcții de transfer, stabilitate, diagrame Bode, regulator PID; Achiziția și Prelucrarea Datelor (TIDS304) — senzori, ADC, interfețe; Proiectarea cu Microprocesoare (TIDS215) — arhitecturi embedded, comunicații industriale.
3.2 de competențe	Cunoașterea modelului matematic al unui sistem de ordinul 1 și 2 și a regulatorului PID; familiarizarea cu limbajul C pentru programare embedded; înțelegerea principiilor de bază ale automatizării (buclă de reglare, senzor, traductor, element de execuție); abilitatea de a utiliza MATLAB/Simulink.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu tablă, videoproiector și acces la internet; platforma Sakai; recomandare: MATLAB cu Simulink și Control System Toolbox instalate înainte de curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator dotat cu: controler logic programabil (PLC Siemens S7-1200 sau echivalent) cu software TIA Portal; modul de simulare a proceselor;

	calculatoare cu MATLAB/Simulink și TIA Portal; osciloscop digital; placă Arduino pentru simularea interfețelor I/O industriale
--	--

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C29. Deține cunoștințe specializate privind teoria sistemelor — sisteme liniare, funcții de transfer, stabilitate, răspuns în frecvență — și aplicarea acestora în modelarea și analiza sistemelor de control și a microsistemelor. C31. Posedă cunoștințe de bază în mecanică și teoria mecanismelor, suficiente pentru înțelegerea principiilor funcționării sistemelor mecatronice, robotice și a componentelor de acționare din sistemele embedded.
5.2 Aptitudini	A29. Modelează și analizează sisteme dinamice utilizând tehnici din teoria sistemelor (funcții de transfer, diagrame Bode, spații de stare), utilizând instrumente software specializate (MATLAB/Simulink) pentru simularea comportamentului sistemelor de control.
5.3 Responsabilitate și autonomie	R29. Își asumă responsabilitatea pentru modelarea corectă a sistemelor dinamice în proiectele ingineresti, validând experimental modelele teoretice și documentând transparent discrepanțele dintre comportamentul simulat și cel real.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Introducere în automatizarea industrială. Definiția sistemului automat: procesul, traductorul, regulatorul, elementul de execuție. Clasificarea sistemelor de reglare: în buclă deschisă vs. închisă. Exemple industriale: controlul temperaturii, al debitului, al nivelului, al presiunii. Ierarhia automatizării: nivelul câmp, control, SCADA, ERP.	2	Prelegere interactivă	
Tema 2: Modelarea proceselor industriale. Modele matematice ale proceselor termice, de debit și de nivel. Funcțiile de transfer de ordinul 1 cu timp mort (FOPDT): constanta de timp T, câștigul K, întârzierea L. Identificarea experimentală prin metoda răspunsului la treaptă: metoda tangentei (Ziegler-Nichols open-loop). Linearizarea în jurul punctului de funcționare.	2	Prelegere interactivă	
Tema 3: Regulatorul PID industrial. Structura PID în forma paralelă, ideală și serială. Efectul fiecărui termen (P, I, D) asupra răspunsului. Filtrarea termenului derivativ. Sintonizarea Ziegler-Nichols (buclă închisă): determinarea perioadei ultime T_u și câștigului ultim K_u . Metoda Cohen-Coon pentru procese FOPDT. Anti-windup: schema back-calculation.	2	Prelegere interactivă	
Tema 4: Controlere logice programabile (PLC). Arhitectura PLC: CPU, module I/O, sursa de alimentare, rack. Ciclul de scanare PLC: citire intrări, execuție program, scriere ieșiri. Limbajele IEC 61131-3: Ladder Diagram (LD), Function Block Diagram (FBD), Structured Text (ST), Instruction List	2	Prelegere interactivă	

(IL), Sequential Function Chart (SFC). Adresarea I/O în Siemens S7-1200.			
Tema 5: Programare PLC — comenzi secvențiale. Logica combinațională în Ladder Diagram: contacte normal deschise/închise, bobine. Temporizatoare: TON (on-delay), TOF (off-delay), TP (pulse). Numărătoare: CTU, CTD, CTUD. Secvențe de comandă: SFC (Sequential Function Chart), pași, tranziții. Programarea unui ciclu automat: start, operație, stop de urgență.	2	Prelegere interactivă	
Tema 6: Controlul continuu cu PLC — blocuri PID. Blocul PID în TIA Portal: configurarea parametrilor, modul manual/automat, tuning automat. Scalarea semnalelor analogice: conversia intrărilor 4–20 mA în unități de proces. Limitarea ieșirii: valoarea minimă și maximă a comenzii. Generatorul de rampă pentru pornirea lină a acționărilor.	2	Prelegere interactivă	
Tema 7: Senzori și traductoare industriale. Senzori de temperatură industriali: termorezistențe PT100/PT1000 (montaj 2, 3, 4 fire), termocupluri (compensare joncțiune rece). Senzori de presiune: piezoresistivi, capacitivi — semnal 4–20 mA. Senzori de debit: debitmetru electromagnetic, cu turbină, cu ultrasunete. Senzori de nivel: cu plutitor, ultrasonic, radar, presostatic.	2	Prelegere interactivă	
Tema 8: Elemente de execuție industriale. Vanele de control: vane cu doi căi și trei căi, caracteristica debit-deschidere (lineară, egală-procentuală). Acționarea vanelor: pneumatică, electrică. Invertoarele de frecvență (VFD): principiul PWM trifazic, parametri de configurare (frecvență minimă/maximă, timp de accelerare/decelerare, protecțiile). Aplicație: controlul vitezei unei pompe.	2	Prelegere interactivă	
Tema 9: Comunicații industriale. Magistrala PROFIBUS DP: arhitectura master-slave, viteza, adresarea. PROFINET: comunicație Ethernet în timp real, sincronizarea. Modbus TCP/RTU: structura cadrului, registrele holding/input, coiluri. OPC-UA: modelul de date, securitatea, serverul și clientul OPC. Integrarea PLC–SCADA prin OPC-UA.	2	Prelegere interactivă	
Tema 10: Sisteme SCADA. Arhitectura SCADA: MTU (Master Terminal Unit), RTU (Remote Terminal Unit), rețeaua de comunicație, stația de operare. Software SCADA (WinCC, Ignition, InTouch): sinopticul dinamic, alarmele, tendințele (trends), rapoartele. Securitatea sistemelor SCADA: vulnerabilități (Stuxnet), segmentarea rețelei, standardul IEC 62443.	2	Prelegere interactivă	
Tema 11: Controlul avansat al proceselor. Controlul în cascadă: bucla primară și	2	Prelegere interactivă	

secundară — avantaje față de PID simplu. Controlul cu decuplare (feedforward): compensarea perturbațiilor măsurabile. Controlul raportului: menținerea raportului dintre două debite. Controlul prin modele interne (IMC): principiul, legătura cu PID. Controlul predictiv bazat pe model (MPC): principiul optimizării pe orizontul de predicție.			
Tema 12: Sisteme de reglare discretă. Eșantionarea și reconstituirea: alegerea perioadei de eșantionare pentru un sistem de reglare. Regulatorul PID discret: forma pozițională și forma incrementală (velocity form). Discretizarea cu metoda Tustin (bilinear). Reglarea digitală directă (DDC): structura, avantajele față de analogic. Exemple de implementare pe PLC.	2	Prelegere interactivă	
Tema 13: Siguranța funcțională și sistemele SIS. Standardul IEC 61511 (siguranță funcțională în procesele industriale). Nivelele de integritate a siguranței (SIL 1–4): calculul probabilității de defectare la cerere (PFD). Sistemul instrumentat de siguranță (SIS): senzor de siguranță, logica de siguranță (PLC de siguranță), elementul de execuție. Funcția instrumentată de siguranță (SIF). Testarea periodică.	2	Prelegere interactivă	
Tema 14: Industria 4.0 și recapitulare. Industria 4.0: cei 4 piloni (IoT industrial, big data, cloud manufacturing, robotică colaborativă). Digital twin al unui sistem de reglare: simularea în timp real a procesului paralel cu procesul real. Edge computing pentru automatizare: procesarea locală, latența. Tranziția de la SCADA clasic la cloud SCADA. Recapitulare generală.	2	Prelegere, recapitulare	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> Ogata, K., Modern Control Engineering, 5th ed., Pearson, 2010. Bolton, W., Programmable Logic Controllers, 6th ed., Newnes, 2015. Bela G. Liptak (ed.), Instrument Engineers' Handbook, Vol. 2: Process Control and Optimization, 4th ed., CRC Press, 2005. <p>Bibliografie complementară</p> <ol style="list-style-type: none"> Seborg, D.E., Edgar, T.F., Mellichamp, D.A., Doyle, F.J., Process Dynamics and Control, 4th ed., Wiley, 2016. Nise, N.S., Control Systems Engineering, 8th ed., Wiley, 2019. <p>Resurse software:</p> <ol style="list-style-type: none"> Siemens TIA Portal Documentation — programare PLC S7-1200/S7-1500: https://support.industry.siemens.com/ MathWorks Simulink Control Design — proiectarea regloarelor: https://www.mathworks.com/help/slcontrol/ Ignition SCADA Documentation — platformă SCADA modernă: https://docs.inductiveautomation.com/ 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Identificarea unui proces și sintonizarea PID în Simulink. Construirea modelului Simulink FOPDT pentru un proces termic simulat. Aplicarea metodei Ziegler-Nichols open-loop pe răspunsul la treaptă.	2	Lucrare practică MATLAB/Simulink	

Implementarea regulatorului PID și compararea P, PI, PID. Analiza efectului anti-windup.			
Lucrarea L2: Programarea PLC — logică combinațională și temporizatoare. Scrierea unui program Ladder Diagram în TIA Portal pentru o linie de asamblare simplă: pornire condiționată, oprire de urgență, temporizatoare TON și TOF. Testarea în simulatorul PLC (PLCSIM). Descărcarea pe PLC fizic și verificarea comportamentului.	2	Lucrare practică TIA Portal	
Lucrarea L3: Programare PLC — secvențe SFC. Implementarea unui ciclu automat în Sequential Function Chart (SFC) cu 5 pași: inițializare, umplere, procesare, golire, așteptare. Definirea tranzițiilor și a condițiilor de trecere. Tratarea stărilor de eroare și a butonului de reset.	2	Lucrare practică TIA Portal	
Lucrarea L4: Controlul PID cu PLC. Configurarea blocului PID_Compact în TIA Portal pentru controlul temperaturii simulate. Scalarea intrării analogice 4–20 mA în grade Celsius. Sintonizarea automată (autotuning) și manuală. Monitorizarea parametrilor în timp real prin panoul de control TIA Portal.	2	Lucrare practică TIA Portal	
Lucrarea L5: Comunicație Modbus TCP între PLC și SCADA. Configurarea serverului Modbus TCP pe PLC-ul Siemens S7-1200. Citirea și scrierea registrelor Holding din software-ul SCADA (Ignition sau Node-RED). Afișarea valorii de proces și a setpoint-ului pe un sinoptic dinamic. Implementarea unei alarme de depășire a pragului.	2	Lucrare practică TIA Portal, SCADA	
Lucrarea L6: Controlul în cascadă în Simulink. Implementarea în Simulink a controlului în cascadă pentru un proces cu două constante de timp: bucla internă (debit) și bucla externă (nivel). Sintonizarea secvențială a celor doi regulatori PID. Compararea perturbației respinse față de controlul PID simplu.	2	Lucrare practică MATLAB/Simulink	
Lucrarea L7: Proiect integrat și colocviu. Implementarea unui sistem automat complet: proces simulat în Simulink + regulator PID sintonizat + comunicație OPC-UA cu sinoptic SCADA minimal. Documentația tehnică: schema buclei de reglare, funcția de transfer identificată, curbele de performanță, compararea simulat vs. experimental. Colocviu oral.	2	Proiect individual, colocviu	
	<p>Bibliografie obligatorie laborator</p> <ol style="list-style-type: none"> Siemens TIA Portal Documentation: https://support.industry.siemens.com/ MathWorks Simulink Control Design Documentation: https://www.mathworks.com/help/slcontrol/ Bolton, W., Programmable Logic Controllers, 6th ed., Newnes, 2015 — exercițiile practice. <p>Bibliografie complementara laborator</p>		

4. Ignition SCADA Documentation: <https://docs.inductiveautomation.com/>
5. Node-RED Documentation — automatizare și SCADA lightweight: <https://nodered.org/docs/>
Resurse online gratuite recomandate
6. Siemens TIA Portal Documentation: <https://support.industry.siemens.com/>
7. MathWorks Simulink: <https://www.mathworks.com/help/simulink/>
8. Ignition SCADA Documentation: <https://docs.inductiveautomation.com/>
9. Node-RED Documentation: <https://nodered.org/docs/>

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

Sistemele Automate acoperă automatizarea industrială de la modelarea și sintonizarea regulatorului PID până la programarea PLC, integrarea SCADA și strategiile avansate de control. Aceste competențe sunt direct aplicabile în industria de proces (chimică, alimentară, energetică), în fabricație și în infrastructura critică.

Componenta de laborator urmărește validarea modelelor de reglare în condiții reale sau simulate și documentarea transparentă a discrepanțelor — direct corelată cu R29 din matricea de corelare, diferențiind această disciplină de cursurile teoretice de teoria sistemelor.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea metodologiei de identificare experimentală FOPDT și a sintonizării PID (Ziegler-Nichols, Cohen-Coon); cunoașterea arhitecturii PLC și a ciclului de scanare; cunoașterea limbajelor IEC 61131-3 (Ladder, SFC, ST); cunoașterea protocoalelor industriale Modbus și PROFINET; cunoașterea structurii SCADA și a principiilor de siguranță funcțională (SIL).	Colocviu oral (20 minute): justificarea deciziilor de proiectare din lucrarea L7 — identificarea procesului, alegerea structurii de reglare, sintonizarea PID, integrarea SCADA — cu o întrebare suplimentară despre un scenariu de automatizare industrială nou	40%
8.5 Seminar/laborator	Funcționalitatea programelor PLC (funcționează corect pe PLCSIM și pe echipamentul fizic, tranzițiile SFC sunt corecte); calitatea configurației PID (procesul urmărește setpoint-ul cu eroare staționar nulă sau sub pragul specificat); calitatea documentației	Evaluare continuă: rapoarte individuale predate pe Sakai după fiecare lucrare (50%) și proiect integrat cu colocviu oral individual la L7 (50%)	60%

	tehnice din L7 (schema buclei, curbele de performanță, validarea față de Simulink); calitatea prezentării la colocviu.		
8.6 Standard minim de performanță:			
Demonstrarea la colocviul din L7 a funcționării buclei de reglare complete (procesul urmărește setpoint-ul cu eroare staționar nulă sau sub 5% din valoarea setpoint-ului) și prezentarea comparației dintre răspunsul obținut și simularea Simulink, cu explicarea cauzei discrepantei. Un sistem de reglare nefuncțional sau o documentație fără secțiunea de validare experimentală nu îndeplinesc standardul minim — această cerință reflectă direct R29 și diferențiază disciplina de un curs teoretic de automatizare.			

Data completării	Semnătura titularului de curs Conf. Dr. Lacrămă Dan Laurențiu	Semnătura titularului de seminar / laborator Conf. Dr. Lacrămă Dan Laurențiu
17.03.2026		
Data avizării în Consiliul Departamentului	Semnătura directorului DIT Lect. dr. Antohe Valerian	
24.03.2026		
Data aprobării în Consiliul Facultății	Semnătura decanului FMDT Conf. dr. Beteringhe Adrian	
07.04.2026		



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei	CLOUD COMPUTING.INTRODUCERE – TIDFAC324						
1.2 Titularul activităților de curs	Conf. Dr. Beteringhe Adrian						
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator	Conf. Dr. Beteringhe Adrian						
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	C	1.8 Regimul disciplinei	FAC

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	3	din care: 2.2 curs	2	2.3 seminar/laborator	0/1
2.4 Total ore din planul de învățământ	42	din care: 2.5 curs	28	2.6 seminar/laborator	0/14
Distribuția fondului de timp					ore
2.7 Activități monitorizate					-
Practică					-
Elaborare proiect diplomă					-
2.8 Activități individuale					33
Documentare					10
Studiu individual					10
Referate					10
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					3
2.9 Total ore pe semestru					75
2.10 Numărul de credite					3

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Rețele de Calculatoare (TIDS301) — stiva TCP/IP, virtualizarea rețelei; Sisteme de Operare (TIDS302) — procese, containere, Linux; Ingineria Programelor (TIDS308) — CI/CD, Git, Docker; Baze de Date (TIDS305) — baze de date distribuite, teorema CAP.
3.2 de competențe	Cunoașterea conceptelor de bază ale virtualizării și ale rețelelor IP; experiența cu linia de comandă Linux; familiarizarea cu Docker la nivel introductiv (pull, run, exec); cunoașterea de bază a Git și a unui pipeline CI/CD.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu videoproiector și acces la internet; platforma Sakai; recomandare: cont AWS Free Tier, Azure for Students sau Google Cloud Free Tier creat înainte de curs; Docker Desktop instalat local.
4.2 de desfășurare a seminarului/laboratorului	Seminar desfășurat în laborator de informatică sau în sala de curs cu acces la internet; conturi cloud gratuite (AWS Free Tier, Google Cloud Free Tier sau Azure for Students); Docker Desktop și kubectl instalate; acces

la un cluster Kubernetes local (Minikube sau Kind) sau la un cluster cloud managed (EKS/GKE/AKS free tier).

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C21. Deține cunoștințe specializate privind arhitecturile cloud computing (IaaS, PaaS, SaaS), containerizarea și principiile DevOps, cu înțelegerea critică a modelelor de livrare continuă a software-ului. C39. Posedă cunoștințe avansate privind arhitecturile și protocoalele sistemelor distribuite și cloud computing — modele de consistență, toleranță la defecte, scalabilitate orizontală — cu înțelegerea critică a compromisurilor teoremei CAP.
5.2 Aptitudini	A21. Proiectează și implementează soluții cloud — arhitecturi IaaS/PaaS/SaaS, containere Docker/Kubernetes, pipeline-uri CI/CD — aplicând principiile DevOps pentru livrarea continuă și fiabilă a software-ului. A22. Planifică și execută migrarea sistemelor și aplicațiilor către infrastructuri cloud, gestionând riscurile tehnice asociate, asigurând continuitatea serviciilor și optimizând costurile operaționale.
5.3 Responsabilitate și autonomie	R11. Își asumă responsabilitatea pentru migrarea și integrarea sistemelor informatice complexe în infrastructuri cloud sau distribuite, gestionând riscurile tehnice asociate și asigurând continuitatea serviciilor pe parcursul tranziției. R34. Gestionează autonom riscurile tehnice și operaționale ale arhitecturilor distribuite și cloud adoptate în proiecte, asumând responsabilitatea pentru deciziile de design privind consistența, disponibilitatea și toleranța la partiții.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Introducere în cloud computing. Definiția NIST a cloud computing-ului: cele 5 caracteristici esențiale, 3 modele de servicii, 4 modele de deployment. Comparăția IaaS, PaaS, SaaS: responsabilitățile furnizorului vs. clientului (shared responsibility model). Marii furnizori: AWS, Azure, Google Cloud — cote de piață, diferențieri. Motivele adoptării cloud-ului: costuri CapEx → OpEx, elasticitate, disponibilitate globală.	2	Prelegere interactivă	
Tema 2: Virtualizarea și containerizarea. Virtualizarea: hypervisor tip 1 (VMware ESXi, KVM) și tip 2 (VirtualBox). Containerizarea vs. mașini virtuale: izolarea, overhead-ul, portabilitatea. Docker: arhitectura (daemon, client, registry), imaginile (layered filesystem, Dockerfile), containerele, volumele, rețelele Docker. Docker Hub și registrele private. Securitatea containerelor: utilizatorul non-root, imaginile minimale (distroless).	2	Prelegere interactivă	
Tema 3: Kubernetes — fundamente. Arhitectura Kubernetes: control plane (API server, etcd, scheduler, controller manager) și nodurile worker (kubelet, kube-proxy). Obiectele Kubernetes: Pod, Deployment, ReplicaSet, Service (ClusterIP, NodePort,	2	Prelegere interactivă	

LoadBalancer), Namespace, ConfigMap, Secret. kubectl: comenzile de bază (apply, get, describe, logs, exec). Manifestele YAML: structura, câmpurile obligatorii.			
Tema 4: Kubernetes — scalare și configurare. Horizontal Pod Autoscaler (HPA): scalarea automată pe baza metricilor CPU/memorie. Persistent Volumes (PV) și Persistent Volume Claims (PVC): stocarea persistentă pentru aplicații stateful. Ingress Controller: rutarea HTTP/HTTPS, certificatele TLS, NGINX Ingress. Helm: managerul de pachete Kubernetes, chart-uri, values.yaml. Actualizările rolling și rollback-ul.	2	Prelegere interactivă	
Tema 5: DevOps și CI/CD. Cultura DevOps: CALMS (Culture, Automation, Lean, Measurement, Sharing). Pipeline CI/CD: etapele (source, build, test, deploy), instrumentele (GitHub Actions, GitLab CI, Jenkins). Strategiile de deployment: rolling update, blue-green, canary. Feature flags: controlul funcționalităților fără deployment. GitOps: ArgoCD, Flux — infrastructura definită în Git.	2	Prelegere interactivă	
Tema 6: Infrastructura ca și cod (IaC). Principiile IaC: idempotență, versionare, auditare. Terraform: provider-i, resurse, state-ul, workspace-urile, modulele. Exemplu: provisionarea unei VM pe AWS cu Terraform. Ansible: playbook-uri, roluri, inventarul, idempotența. Comparatie Terraform (provisionare) vs. Ansible (configurare). Pulumi: IaC cu limbaje de programare general-purpose.	2	Prelegere interactivă	
Tema 7: Servicii cloud AWS — compute și stocare. EC2: tipurile de instanțe (familiile T, M, C, R), AMI-urile, grupurile de securitate, Elastic IP. Auto Scaling Groups: politicile de scalare (target tracking, step scaling). S3: bucket-uri, obiecte, clase de stocare (Standard, IA, Glacier), politicile de bucket, versioning, lifecycle rules. EBS vs. EFS vs. S3: când se folosește fiecare.	2	Prelegere interactivă	
Tema 8: Servicii cloud AWS — rețea și baze de date. VPC: subrețele publice/private, route tables, Internet Gateway, NAT Gateway, Security Groups vs. NACLs. RDS: motoarele suportate, Multi-AZ (disponibilitate ridicată), Read Replicas (scalare citire), backup automat. DynamoDB: modelul cheie-valoare și document, partitionarea, consistența eventuală vs. puternică, capacitatea on-demand vs. provisioned. ElastiCache: Redis și Memcached.	2	Prelegere interactivă	
Tema 9: Arhitecturi serverless. AWS Lambda: modelul event-driven, handler-ul, contextul,	2	Prelegere interactivă	

<p>limitele (timeout, memorie, dimensiunea pachetului). Declanșatorii Lambda: API Gateway, S3, DynamoDB Streams, SQS, EventBridge. API Gateway: REST API vs. HTTP API, autentificarea (Cognito, Lambda Authorizer). Arhitectura serverless pentru aplicații web: comparația cu arhitectura containerizată — cold start, costuri, debugging.</p>			
<p>Tema 10: Sisteme distribuite — fundamente. Teorema CAP: consistență, disponibilitate, toleranță la partiții — imposibilitatea simultană. Sistemele CP (HBase, ZooKeeper), AP (Cassandra, DynamoDB) și CA (teoretic). Consistența eventuală: BASE vs. ACID. Modelele de consistență: linearizabilitate, consistența cauzală, consistența sesiunii. Protocoale de consens distribuit: Raft (principiu), aplicarea în etcd.</p>	2	Prelegere interactivă	
<p>Tema 11: Observabilitate și monitorizare cloud. Pilonii observabilității: metrici, loguri, trace-uri distribuite. AWS CloudWatch: metrici, alerte, dashboard-uri, log groups. Prometheus și Grafana: arhitectura, scraping-ul, PromQL de bază, dashboard-uri. Jaeger: trace-ul distribuit, span-urile, propagarea contextului. SLI, SLO, SLA: definițiile, error budgets (principiile SRE). Alertarea: runbook-urile, escaladarea.</p>	2	Prelegere interactivă	
<p>Tema 12: Securitatea în cloud. Modelul shared responsibility în detaliu: ce securizează furnizorul vs. clientul. IAM (Identity and Access Management): principiul least privilege, roluri, politici, grupuri, MFA. Secretele în cloud: AWS Secrets Manager vs. HashiCorp Vault. Conformitatea: SOC 2, ISO 27001, GDPR în cloud (localizarea datelor, dreptul la ștergere). Securitatea containerelor: imagine scanning (Trivy), runtimesecurity (Falco), network policies Kubernetes.</p>	2	Prelegere interactivă	
<p>Tema 13: Optimizarea costurilor cloud. Modelele de prețuri AWS: On-Demand, Reserved (1/3 ani), Spot Instances, Savings Plans. Instrumentele de cost: AWS Cost Explorer, AWS Budgets, AWS Trusted Advisor. Strategii de optimizare: right-sizing, eliminarea resurselor nefolosite, arhivarea în Glacier, transferul de date (egress costs). FinOps: cultura și procesele de optimizare a costurilor cloud în echipe.</p>	2	Prelegere interactivă	
<p>Tema 14: Tendințe și recapitulare. Multi-cloud și hybrid cloud: motivele, provocările de management (Anthos, Azure Arc). Edge computing: AWS Outposts, CloudFront, Wavelength. Kubernetes la scară: serviciile managed (EKS, GKE, AKS) vs. self-managed.</p>	2	Prelegere, recapitulare	

Platformele de tip PaaS pentru dezvoltatori: Vercel, Railway, Render — abstractizarea infrastructurii. AI/ML în cloud: SageMaker, Vertex AI, Azure ML. Recapitulare generală.			
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> 1. Kleppmann, M., Designing Data-Intensive Applications, O'Reilly Media, 2017. 2. Burns, B., Beda, J., Hightower, K., Kubernetes: Up and Running, 3rd ed., O'Reilly Media, 2022. 3. Kim, G., Humble, J., Behr, P., Forsgren, N., The DevOps Handbook, 2nd ed., IT Revolution Press, 2021. <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Tanenbaum, A.S., Van Steen, M., Distributed Systems: Principles and Paradigms, 3rd ed., Pearson, 2017. Disponibil gratuit la: https://www.distributed-systems.net/ 5. AWS Documentation — referință completă a serviciilor AWS: https://docs.aws.amazon.com/ <p>Resurse software:</p> <ol style="list-style-type: none"> 6. AWS Documentation: https://docs.aws.amazon.com/ 7. Kubernetes Documentation: https://kubernetes.io/docs/ 8. Terraform Documentation (HashiCorp): https://developer.hashicorp.com/terraform/docs 		
6.2 Seminar	Nr. ore	Metode de predare	Observații
Seminarul S1: Crearea și rularea containerelor Docker. Instalarea Docker Desktop. Rularea containerului hello-world. Construirea unei imagini Docker cu Dockerfile simplu (aplicație Python Flask sau Node.js). Publicarea imaginii pe Docker Hub. Rularea cu variabile de mediu și volume montate.	1	Seminar practic	
Seminarul S2: Docker Compose — aplicații multi-container. Scrierea unui fișier docker-compose.yml pentru o aplicație cu 3 servicii: backend, bază de date și frontend. Rețele interne Docker Compose. Variabile de mediu din fișier .env. Comenzile docker compose up, down, logs, exec.	1	Seminar practic	
Seminarul S3: Kubernetes — primii pași. Configurarea unui cluster local cu Minikube sau Kind. Crearea unui Deployment și a unui Service pentru o aplicație containerizată. Scalarea manuală cu kubectl scale. Actualizarea imaginii (rolling update) și rollback.	1	Seminar practic	
Seminarul S4: Kubernetes — ConfigMap, Secret și Ingress. Stocarea configurației în ConfigMap și a credențialelor în Secret. Montarea lor în Pod ca variabile de mediu sau volume. Configurarea unui Ingress Controller NGINX. Routarea traficului HTTP la două servicii diferite pe baza path-ului.	1	Seminar practic	
Seminarul S5: Pipeline CI/CD cu GitHub Actions. Crearea unui workflow GitHub Actions care: compilează/testează aplicația, construiește imaginea Docker, o publică pe Docker Hub sau GitHub Container Registry, o deploaează în clusterul Kubernetes (kubectl apply). Secretele în GitHub Actions.	1	Seminar practic	
Seminarul S6: Terraform — provisionarea infrastructurii AWS. Scrierea unui fișier main.tf care creează o instanță EC2, un	1	Seminar practic AWS	

Security Group și o cheie SSH pe AWS Free Tier. Comenzile terraform init, plan, apply, destroy. Modificarea infrastructurii și observarea planului de schimbări.			
Seminarul S7: AWS S3 și IAM. Crearea unui bucket S3 cu politici de acces. Uploadul și downloadul obiectelor cu AWS CLI. Configurarea unui utilizator IAM cu permisiuni minime (least privilege). Crearea unui rol IAM pentru o instanță EC2. Activarea MFA pentru contul AWS.	1	Seminar AWS	practic
Seminarul S8: AWS Lambda și API Gateway. Scrierea și deployarea unei funcții Lambda (Python sau Node.js) care procesează un eveniment JSON. Configurarea unui API Gateway HTTP API cu o rută GET/POST legată la Lambda. Testarea din browser și cu curl. Monitorizarea cu CloudWatch Logs.	1	Seminar AWS	practic
Seminarul S9: Observabilitate — Prometheus și Grafana. Instalarea Prometheus și Grafana în clusterul Kubernetes cu Helm (kube-prometheus-stack). Explorarea metricilor de bază (CPU, memorie, request rate). Crearea unui dashboard Grafana pentru aplicația proprie. Configurarea unei alerte pe baza unei condiții PromQL.	1	Seminar Kubernetes	practic
Seminarul S10: Securitatea containerelor. Scanarea vulnerabilităților dintr-o imagine Docker cu Trivy. Aplicarea unui Network Policy Kubernetes pentru restricționarea traficului inter-pod. Rularea containerului ca utilizator non-root. Auditarea permisiunilor IAM cu AWS IAM Access Analyzer.	1	Seminar	practic
Seminarul S11: Baze de date în cloud. Crearea unei instanțe RDS PostgreSQL pe AWS Free Tier. Conectarea la baza de date din aplicație cu connection string din Secrets Manager. Configurarea unui Read Replica. Explorarea DynamoDB: crearea unui tabel, scrierea și citirea elementelor cu AWS SDK Python (boto3).	1	Seminar AWS	practic
Seminarul S12: Optimizarea costurilor cloud. Analiza costurilor contului AWS Free Tier cu Cost Explorer. Identificarea resurselor neutilizate cu Trusted Advisor. Calcularea economiilor prin Reserved Instances vs. On-Demand pentru un scenariu dat. Exercițiu de estimare a costului lunar al unei arhitecturi cu AWS Pricing Calculator.	1	Seminar AWS	practic
Seminarul S13: Proiect integrat — arhitectură cloud completă. Proiectarea și implementarea în echipă (2-3 studenți) a unei aplicații web cu: container Docker, deployment Kubernetes, pipeline CI/CD GitHub Actions, infrastructură provisionată cu Terraform și monitorizare cu	1	Proiect în echipă	

Prometheus. Prezentarea arhitecturii în diagramă C4.			
Seminarul S14: Prezentarea proiectelor și evaluare. Fiecare echipă prezintă arhitectura și demonstrează live funcționarea. Discuție despre compromisurile tehnice alese (CAP theorem, costul vs. disponibilitatea). Feedback colectiv. Autoevaluarea competențelor dobândite.	1	Prezentări, evaluare	
<p>Bibliografie obligatorie semianr</p> <ol style="list-style-type: none"> 1. Kubernetes Documentation: https://kubernetes.io/docs/ 2. AWS Documentation: https://docs.aws.amazon.com/ 3. Burns, B. et al., Kubernetes: Up and Running, 3rd ed., O'Reilly Media, 2022. <p>Bibliografie complementara semianr</p> <ol style="list-style-type: none"> 4. Terraform Documentation: https://developer.hashicorp.com/terraform/docs 5. GitHub Actions Documentation: https://docs.github.com/en/actions <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. Kubernetes Documentation: https://kubernetes.io/docs/ 7. AWS Documentation: https://docs.aws.amazon.com/ 8. Terraform Documentation: https://developer.hashicorp.com/terraform/docs 9. Play with Kubernetes — laborator online gratuit: https://labs.play-with-k8s.com/ 			

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

Cloud Computing acoperă competențele cele mai solicitate pe piața muncii TI din România și Europa: containerizarea cu Docker și Kubernetes, pipeline-urile CI/CD, infrastructura ca și cod cu Terraform și serviciile cloud AWS. Disciplina formează direct A21 și A22 — proiectarea soluțiilor cloud și migrarea aplicațiilor — prin practicarea lor în cele 14 seminarii. Înțelegerea compromisurilor teoremei CAP și asumarea deciziilor de design cloud (C39, R34) diferențiază inginerul cloud de simplul utilizator de servicii cloud.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea modelelor IaaS/PaaS/SaaS și a shared responsibility model; cunoașterea arhitecturii Kubernetes (control plane, obiecte principale); cunoașterea principiilor DevOps și a etapelor unui pipeline CI/CD; cunoașterea teoremei CAP și a sistemelor CP vs. AP; cunoașterea principalelor servicii AWS (EC2, S3, RDS, Lambda, VPC, IAM); cunoașterea principiului IaC cu Terraform.	Colocviu oral (20 minute): justificarea deciziilor de arhitectură din proiectul integrat (alegerea serviciilor cloud, compromisurile CAP, strategia de deployment, costul estimat) cu o întrebare suplimentară despre un scenariu de migrare cloud nou	40%
8.5 Seminar/laborator	Funcționalitatea proiectului integrat din S13-S14 (aplicația rulează în Kubernetes, pipeline-ul CI/CD se	Evaluare continuă: implementări individuale predate pe Sakai după seminarele S1-S12 (50%) și proiect integrat în echipă cu prezentare live la S14 (50%)	60%

	<p>execută fără erori, infrastructura Terraform este provisionabilă cu terraform apply); calitatea implementărilor din seminarii (comenzile funcționează, configurațiile sunt corecte); calitatea prezentării arhitecturii în diagramă C4 și a justificărilor tehnice.</p>		
--	--	--	--

8.6 Standard minim de performanță:

Demonstrarea la S14 a unui pipeline CI/CD funcțional care deployează automat o aplicație containerizată într-un cluster Kubernetes la fiecare push pe branch-ul main, cu cel puțin etapele build, push image și kubectl apply. Un proiect fără pipeline funcțional sau fără deployment în Kubernetes nu îndeplinește standardul minim, indiferent de calitatea celorlalte componente.

Data completării	Semnătura titularului de curs Conf. Dr. Beteringhe Adrian	Semnătura titularului de seminar / laborator Conf. Dr. Beteringhe Adrian
16.03.2026		
Data avizării în Consiliul Departamentului	Semnătura directorului DIT Lect. Dr. Antohe Valerian	
24.03.2026		
Data aprobării în Consiliul Facultății	Semnătura decanului FMDT Conf. Dr. Beteringhe Adrian	
07.04.2026		



UNIVERSITATEA INTERNAȚIONALĂ DANUBIUS
FACULTATEA DE MEDIA, DESIGN ȘI TEHNOLOGIE
DEPARTAMENTUL: INFORMATICĂ ȘI TEHNOLOGII
DOMENIUL DE STUDII DE LICENȚĂ: CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI
PROGRAMUL DE STUDII: TEHNOLOGIA INFORMAȚIEI
STUDII UNIVERSITARE DE LICENȚĂ CU FRECVENȚĂ
SERIA: 2026-2030
ANUL UNIVERSITAR: 2028-2029

1. Date despre disciplină

1.1 Denumirea disciplinei		INSTRUIRE ASISTATĂ DE CALCULATOR – TIDFAC325					
1.2 Titularul activităților de curs		Conf. Dr. Beteringhe Adrian					
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator		Conf. Dr. Beteringhe Adrian					
1.5 Anul de studiu	III	1.6 Semestrul	1	1.7 Tipul de evaluare	E	1.8 Regimul disciplinei	FAC

2. Timpul total estimat (ore pe semestru al activităților didactice)

2.1 Număr de ore pe săptămână	2	din care: 2.2 curs	1	2.3 seminar/laborator	0/1
2.4 Total ore din planul de învățământ	28	din care: 2.5 curs	14	2.6 seminar/laborator	0/14
Distribuția fondului de timp					ore
2.7 Activități monitorizate					-
Practică					-
Elaborare proiect diplomă					-
2.8 Activități individuale					22
Documentare					7
Studiu individual					7
Referate					6
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					2
2.9 Total ore pe semestru					50
2.10 Numărul de credite					2

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	
3.2 de competențe	Familiarizarea cu platforma Sakai sau Moodle ca student; capacitatea de a crea documente structurate (Word, Google Docs); cunoașterea de bază a dreptului de autor și a licențelor Creative Commons; experiența utilizării instrumentelor multimedia (înregistrare audio-video, editare de bază).

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu videoprojector și acces la internet; platforma Sakai a universității; recomandare: cont sandbox Moodle (moodle.org) sau instalare locală creată înainte de primul curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator de informatică cu calculatoare individuale și acces la internet; platformă Moodle (instalare locală sau sandbox cloud); H5P Editor online pentru conținut interactiv; OBS Studio sau Loom pentru screencasting; studenții pot folosi propriul laptop.

5. Competențele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei sub forma tabelului de mai jos:

5.1 Cunoștințe	C40. Cunoaște principiile și metodologiile instruirii asistate de calculator (IAC) și ale e-learning-ului, inclusiv standardele internaționale de conținut educațional digital (SCORM, xAPI) și platformele LMS.
5.2 Aptitudini	A35. Elaborează și implementează cursuri și materiale de instruire asistată de calculator, utilizând platforme LMS și standarde e-learning (SCORM), adaptând conținutul la nevoile specifice ale diferitelor categorii de cursanți.
5.3 Responsabilitate și autonomie	R35. Acționează responsabil în rolul de formator sau tutore în activități de instruire asistată de calculator, asigurând calitatea conținutului educațional digital, respectarea drepturilor de autor și accesibilitatea cursurilor pentru toți beneficiarii.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Introducere în IAC și teorii ale învățării. Definiția IAC și evoluția e-learning-ului (1.0, 2.0, 3.0). Avantajele și limitele față de instruirea tradițională. Teorii ale învățării aplicate în e-learning: behaviorism, cognitivism, constructivism. Principiile multimedia Mayer: reducerea încărcăturii cognitive. Modelul ADDIE: etapele proiectării instrucționale.	2	Prelegere interactivă	
Tema 2: Platforme LMS și standarde e-learning. LMS — definiție, roluri, criterii de selecție. Moodle, Canvas, Google Classroom: comparație. Platforme MOOC: Coursera, edX. Standardele SCORM 1.2 și 2004: structura pachetului, comunicarea cu LMS-ul. xAPI (Tin Can API): modelul actor-verb-obiect, Learning Record Store. Alegerea standardului în funcție de scenariu.	2	Prelegere interactivă	
Tema 3: Designul conținutului educațional digital. Obiectivele de învățare SMART și taxonomia Bloom revizuită. Chunking-ul și secvențializarea conținutului. Tipurile de interactivitate (Allen): pasivă, limitată, moderată, completă. Conținut multimedia: video educațional (durata optimă, subtitluri), audio, infografice. Accesibilitatea conținutului: standardul WCAG 2.1.	2	Prelegere interactivă	
Tema 4: Evaluarea și gamificarea în e-learning. Evaluarea formativă și sumativă în LMS: quiz cu feedback automat, assignment, workshop peer review. Prevenirea fraudei: itemi randomizați, limite de timp. Gamificarea: puncte, insigne, clasamente, niveluri — avantaje și riscuri. Implementarea în Moodle: pluginul Level Up!, badges. Motivația intrinsecă vs. extrinsecă.	2	Prelegere interactivă	
Tema 5: Învățarea adaptivă și learning analytics. Sistemele de tutoriat inteligent (ITS): modelul domeniului, al studentului, al pedagogiei. Learning analytics: metrice de	2	Prelegere interactivă	

bază (completion rate, time-on-task, scor mediu). xAPI ca sursă de date pentru analytics. Personalizarea conținutului pe baza datelor de parcurs. Implicații etice: confidențialitatea datelor studenților.			
Tema 6: Colaborare online și mobile learning. Comunitățile de practică online (Wenger). Forumuri, wiki-uri și webinare în Moodle: BigBlueButton, bune practici de facilitare. Mobile learning: proiectarea pentru ecrane mici. Microlearning: conținut sub 5 minute, formatele, integrarea în fluxul de lucru. Notificările push ca instrument educațional.	2	Prelegere interactivă	
Tema 7: Drept de autor, calitate și tendințe. Licențele Creative Commons: tipurile, alegerea licenței corecte. Resursele educaționale deschise (OER): repositoriile principale (MERLOT, OER Commons). Standardele de calitate: Quality Matters, European e-Competence Framework. AI în educație: ChatGPT, tutori AI, generarea automată de conținut — oportunități și riscuri etice. Recapitulare generală.	2	Prelegere, recapitulare	

Bibliografie obligatorie

1. Horton, W., e-Learning by Design, 2nd ed., Pfeiffer, 2011.
2. Clark, R.C., Mayer, R.E., e-Learning and the Science of Instruction, 4th ed., Wiley, 2016.

3. Moodle Documentation: <https://docs.moodle.org/>

Bibliografie complementară

4. Siemens, G., Connectivism: A Learning Theory for the Digital Age, 2005. Disponibil la: https://www.itdl.org/Journal/Jan_05/article01.htm
5. ADL Initiative — xAPI și standardele e-learning: <https://adlnet.gov/>

Resurse software:

6. Moodle Documentation: <https://docs.moodle.org/>
7. H5P Documentation — conținut educațional interactiv: <https://h5p.org/documentation>
8. ADL xAPI Documentation: <https://adlnet.gov/projects/xapi/>

6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Explorarea Moodle și structurarea unui curs. Navigarea în Moodle (sandbox sau instalare locală): crearea unui curs, adăugarea secțiunilor, resurselor (page, file, URL) și activităților (forum, quiz, assignment). Configurarea rolurilor. Redactarea obiectivelor de învățare SMART pentru 3 module ale unui mini-curs pe un subiect tehnic ales.	2	Lucrare practică Moodle	
Lucrarea L2: Conținut interactiv — H5P și multimedia. Crearea unui Interactive Video H5P cu 3 întrebări la momente cheie. Crearea unui Course Presentation cu 5 slide-uri interactive și a unui Drag and Drop. Înregistrarea unui screencast de 3-5 minute cu OBS Studio sau Loom pe un subiect tehnic. Adăugarea subtitrurilor. Integrarea tuturor în cursul Moodle.	2	Lucrare practică H5P, OBS	
Lucrarea L3: Evaluarea online în Moodle. Configurarea unui Quiz cu 10 întrebări de	2	Lucrare practică Moodle	

tipuri diferite (multiple choice, matching, short answer, numerical) din o bancă de întrebări, cu randomizare și feedback automat per întrebare. Configurarea unui Assignment cu rubrică de evaluare (3 criterii). Configurarea unui Workshop (peer review) cu 2 criterii și testarea cu un coleg.			
Lucrarea L4: Gamificare și condiționarea activităților. Instalarea și configurarea pluginului Level Up! (sau sandbox preinstalat): atribuirea de puncte pentru parcurgerea paginilor, postarea în forum și finalizarea quiz-urilor. Crearea a 3 insigne (badges) cu criterii automate de acordare. Configurarea activităților condiționate (completion conditions) pentru a crea un parcurs secvențial.	2	Lucrare practică Moodle	
Lucrarea L5: SCORM și learning analytics. Crearea unui pachet SCORM simplu din conținut H5P sau cu Adapt Framework și importul în Moodle. Testarea comunicării LMS: completion tracking, scor transmis. Explorarea rapoartelor Moodle: completion report, activity report. Calculul manual al metricilor: rata de finalizare, scorul mediu per quiz. Discuția implicațiilor etice privind datele studenților.	2	Lucrare practică Moodle, SCORM	
Lucrarea L6: Accesibilitate și resurse educaționale deschise. Verificarea accesibilității conținutului Moodle cu WAVE (wave.webaim.org): identificarea și corectarea a 3 probleme (contrast, text alternativ, structura titlurilor). Identificarea a 5 resurse OER de pe OER Commons sau MERLOT utilizabile în cursul propriu. Atribuirea corectă a licenței CC pentru fiecare resursă. Marcarea unui conținut propriu cu licență CC.	2	Lucrare practică, OER	
Lucrarea L7: Proiect integrat și prezentare. Finalizarea cursului Moodle proiectat pe parcursul semestrului: minim 3 module cu conținut text + multimedia, video screencast integrat, conținut H5P, quiz cu bancă de întrebări, assignment cu rubrică și badge de finalizare. Testarea cursului cu un coleg în rolul de student. Prezentarea live a cursului: demonstrarea funcționalităților, justificarea deciziilor pedagogice, licențierea resurselor.	2	Proiect individual, prezentare	
	<p>Bibliografie obligatorie laborator</p> <ol style="list-style-type: none"> 1. Moodle Documentation: https://docs.moodle.org/ 2. H5P Documentation: https://h5p.org/documentation 3. Clark, R.C., Mayer, R.E., e-Learning and the Science of Instruction, 4th ed., Wiley, 2016. <p>Bibliografie complementara laborator</p> <ol style="list-style-type: none"> 4. ADL xAPI Documentation: https://adlnet.gov/projects/xapi/ 5. OER Commons: https://www.oercommons.org/ <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. Moodle Documentation: https://docs.moodle.org/ 		

7. H5P Documentation: <https://h5p.org/documentation>
 8. OER Commons: <https://www.oercommons.org/>
 9. WAVE Web Accessibility Evaluation Tool: <https://wave.webaim.org/>

7. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității și asociațiilor profesionale, angajatori reprezentativi din domeniul aferent programului

Instruirea Asistată de Calculator formează competențe direct aplicabile: orice inginer software care lucrează cu platforme de onboarding, documentație interactivă sau sisteme de training intern aplică principiile IAC. Construirea unui curs Moodle complet — de la designul instrucțional la licențierea resurselor și testarea accesibilității — formează direct A35 și R35.

Cerința accesibilității și a licențierii corecte a resurselor OER din L6 și L7 formează responsabilitatea R35: calitatea conținutului digital, respectarea dreptului de autor și asigurarea accesibilității pentru toți beneficiarii.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea principiilor designului instrucțional (ADDIE, taxonomia Bloom, principiile Mayer); cunoașterea standardelor SCORM și xAPI și a rolului lor în comunicarea LMS-conținut; cunoașterea tipologiei LMS-urilor și a criteriilor de selecție; cunoașterea principiilor gamificării și a learning analytics; cunoașterea cadrului dreptului de autor și al licențelor Creative Commons.	Colocviu oral (20 minute): prezentarea și justificarea deciziilor pedagogice din cursul Moodle finalizat — alinierea obiectivelor cu activitățile și evaluările, alegerea tipurilor de conținut, licențierea resurselor, soluțiile de accesibilitate aplicate — cu o întrebare suplimentară din materia de curs	60%
8.5 Seminar/laborator	Funcționalitatea și completitudinea cursului Moodle finalizat (minim 3 module, video screencast, conținut H5P, quiz cu bancă de întrebări, assignment cu rubrică, badge de finalizare — toate funcționale); corectitudinea licențierii resurselor utilizate (L6); calitatea prezentării live din L7.	Evaluare continuă: livrabile individuale predate pe Sakai după lucrările L2, L3, L5, L6 (50%) și cursul Moodle complet cu prezentare live la L7 (50%)	40%

8.6 Standard minim de performanță:

Predarea unui curs Moodle funcțional parcurgibil complet de un student de test, care conține cel puțin: un obiectiv de învățare SMART per modul, un element H5P interactiv, un quiz cu minim 5 întrebări cu feedback automat și cel puțin o resursă externă cu licența Creative Commons corect atribuită. Un curs care nu poate fi parcurs de la capăt sau care utilizează resurse fără atribuire de licență nu îndeplinește standardul minim.

Data completării 16.03.2026	Semnătura titularului de curs Conf. Dr. Beteringhe Adrian	Semnătura titularului de seminar / laborator Conf. Dr. Beteringhe Adrian
Data avizării în Consiliul Departamentului 24.03.2026	Semnătura directorului DIT Lect. Dr. Antohe Valerian	
Data aprobării în Consiliul Facultății 07.04.2026	Semnătura decanului FMDT Conf. Dr. Beteringhe Adrian	