



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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei	PROTOCOALE DE COMUNICAȚII – TIDS401						
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	IV	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					69
Documentare					20
Studiu individual					21
Referate					20
Teme casă					5
Proiect					
Alte activități (inclusive colocviu, examen)					3
2.9 Total ore pe semestru					125
2.10 Numărul de credite					5

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Rețele de Calculatoare (TIDS301); Sisteme și Rețele de Comunicații (TIDS313); Rețele Locale de Calculatoare (TIDS312).
3.2 de competențe	Cunoașterea stivei TCP/IP și a modelului OSI; familiarizarea cu Wireshark și cu linia de comandă Linux pentru diagnosticarea rețelelor.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu videoproiector și acces la internet; platforma Sakai.
4.2 de desfășurare a seminarului/laboratorului	Laborator de rețele cu switch-uri și routere fizice sau acces la GNS3/Packet Tracer; Wireshark instalat; calculatoare cu Linux.

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

5.1 Cunoștințe	C23. Cunoaște în profunzime protocoalele avansate de comunicații și arhitecturile sistemelor de programare pentru timp real, cu conștientizarea critică a cerințelor de performanță și fiabilitate în sisteme critice.
----------------	--

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. R23. Își asumă responsabilitatea pentru înțelegerea corectă și completă a cerințelor beneficiarilor, interacționând proactiv cu utilizatorii și factorii de decizie pentru clarificarea ambiguităților și validarea soluțiilor propuse pe parcursul întregului proces de dezvoltare.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Recapitulare TCP/IP și modelul OSI. Stiva TCP/IP vs. OSI: corespondența straturilor, encapsularea datelor. Adresarea IP, subnetting VLSM, ARP, ICMP. Analiza unui pachet complet cu Wireshark.	3	Prelegere interactivă	
Tema 2: TCP avansat. Controlul congestiei: CUBIC, BBR, algoritmul slow start și congestion avoidance. SACK (Selective Acknowledgment) și TCP Fast Open. Analiza performanței TCP prin măsurători RTT și throughput.	3	Prelegere interactivă	
Tema 3: QUIC și HTTP/3. Motivatia QUIC: eliminarea head-of-line blocking din TCP. Multiplexarea stream-urilor, 0-RTT connection resumption, integrarea TLS 1.3. Comparatie TCP+HTTP/2 vs. QUIC+HTTP/3.	3	Prelegere interactivă	
Tema 4: Rutarea dinamică avansată — OSPF. OSPF multi-arie: aria backbone (0), ariile stub și NSSA. LSA-urile (tip 1-5), baza de date LSDB, algoritmul Dijkstra. Configurarea și verificarea adiacerilor OSPF.	3	Prelegere interactivă	
Tema 5: BGP — protocolul de rutare inter-AS. Sesiunile eBGP și iBGP, atributele (AS-PATH, NEXT-HOP, LOCAL_PREF, MED). Filtrarea prefixelor cu prefix-list și route-map. Rolul BGP în funcționarea Internetului.	3	Prelegere interactivă	
Tema 6: MPLS și QoS. Label switching: tabela LFIB, LSP-urile, operațiile push/swap/pop. Traffic engineering cu MPLS-TE: rezervarea lățime de bandă. DiffServ: DSCP, PHB-urile (EF, AF, BE), mecanismele de coadă.	3	Prelegere interactivă	
Tema 7: IPv6 avansat. Structura header-ului IPv6, tipurile de adrese (GUA, LLA, anycast, multicast). NDP (Neighbor Discovery Protocol): RS, RA, NS, NA. Autoconf SLAAC și DHCPv6. Mecanisme de tranzitie IPv4/IPv6: dual-stack, tunneling, NAT64.	3	Prelegere interactivă	
Tema 8: TLS 1.3 și securitatea transport. Handshake-ul TLS 1.3: mesajele ClientHello, ServerHello, Finished, reducerea la 1-RTT.	3	Prelegere interactivă	

Suite-le de cifrare, certificatele X.509, OCSP Stapling. Atacuri: downgrade, MITM, certificate pinning.			
Tema 9: IPsec si VPN. Modurile transport si tunel, protocoalele ESP si AH. IKEv2: fazele de negociere, autentificarea, Perfect Forward Secrecy. Site-to-site VPN vs. Remote Access VPN. Comparatie IPsec vs. WireGuard.	3	Prelegere interactiva	
Tema 10: Protocoale pentru aplicatii distribuite. gRPC: Protocol Buffers, streaming unar si bidirectional, comparatie cu REST. WebSocket: handshake-ul HTTP Upgrade, full-duplex, cazuri de utilizare. MQTT: publish-subscribe, nivelele QoS 0/1/2, aplicatii IoT.	3	Prelegere interactiva	
Tema 11: Protocoale pentru timp real. RTP/RTCP: structura pachetelor, SSRC, jitter buffer, rapoartele SR/RR. SIP: dialog-ul, SDP, semnalizarea apelurilor VoIP. WebRTC: ICE, STUN, TURN, negocierea peer-to-peer.	3	Prelegere interactiva	
Tema 12: Protocoale industriale si embedded. OPC-UA: modelul de date ierarhic, securitatea, profilurile de comunicatie. Modbus TCP: structura PDU, adresarea registrelor, limitarile protocolului. CAN bus: arbitrarea, structura cadrului, utilizarea in automotive si automatizari.	3	Prelegere interactiva	
Tema 13: Protocoale pentru retele de mare viteza. RDMA (InfiniBand, RoCE v2): zero-copy, kernel bypass, utilizarea in HPC si data center. DPDK: procesarea pachetelor in spatiul utilizator. Protocoale pentru 5G: PDCP, SDAP, GTP-U in planul utilizator.	3	Prelegere interactiva	
Tema 14: Protocoale IoT si recapitulare. CoAP: REST peste UDP, metodele GET/PUT/POST/DELETE, observarea resurselor. LwM2M: modelul obiect-instanta-resursa, bootstrapping. LoRaWAN: structura retelei, clasele A/B/C, ADR. Recapitulare generala.	3	Prelegere, recapitulare	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> 1. Tanenbaum, A.S., Feamster, N., Wetherall, D., Computer Networks, 6th ed., Pearson, 2021. 2. Kurose, J.F., Ross, K.W., Computer Networking: A Top-Down Approach, 8th ed., Pearson, 2021. 3. RFC-urile relevante: https://www.rfc-editor.org/ <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Forouzan, B.A., Data Communications and Networking, 5th ed., McGraw-Hill, 2013. 5. IETF Datatracker: https://datatracker.ietf.org/ <p>Resurse software:</p> <ol style="list-style-type: none"> 6. Wireshark Documentation: https://www.wireshark.org/docs/ 7. IETF RFC Editor: https://www.rfc-editor.org/ 8. GNS3 Documentation: https://docs.gns3.com/ 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Analiza TCP si QUIC cu Wireshark. Capturarea traficului HTTP/2 (TCP) si HTTP/3 (QUIC) pe acelasi site. Identificarea mecanismelor de control al	2	Lucrare practica Wireshark	

congestiei si compararea latentei primului byte.			
Lucrarea L2: Configurarea OSPF multi-arie in GNS3. Configurarea unui scenariu cu aria backbone si doua arii stub. Verificarea adiacerilor, a bazei LSDB si a rutelor instalate. Simularea caderii unui link si observarea reconvergentei.	2	Lucrare practica GNS3	
Lucrarea L3: Configurarea BGP in GNS3. Configurarea a trei AS-uri cu sesiuni eBGP si iBGP. Aplicarea unui filtru de prefix si a atributului LOCAL_PREF. Verificarea tabelii BGP si a rutei selectate.	2	Lucrare practica GNS3	
Lucrarea L4: Implementare gRPC si MQTT in Python. Implementarea unui serviciu gRPC cu Protocol Buffers. Configurarea unui broker Mosquitto si a unui client Python cu QoS 1. Compararea latentei gRPC vs. REST.	2	Lucrare practica Python	
Lucrarea L5: Analiza TLS 1.3 si IPsec cu Wireshark. Capturarea unui handshake TLS 1.3 si identificarea mesajelor principale. Configurarea unui tunel IPsec site-to-site simplu si verificarea encapsularii ESP.	2	Lucrare practica Wireshark	
Lucrarea L6: Flux RTP si analiza jitter. Generarea unui flux RTP simulat cu ffmpeg. Masurarea jitter-ului cu RTP Stream Analysis din Wireshark. Configurarea unui jitter buffer si observarea efectului asupra calitatii.	2	Lucrare practica Wireshark, ffmpeg	
Lucrarea L7: Proiect integrat si colocviu. Implementarea unui scenariu end-to-end care combina doua protocoale din laborator. Documentarea alegerii protocoalelor cu justificare tehnica comparativa. Colocviu oral.	2	Proiect individual, colocviu	
	<p>Bibliografie obligatorie laborator</p> <p>1. Wireshark Documentation: https://www.wireshark.org/docs/</p> <p>2. GNS3 Documentation: https://docs.gns3.com/</p> <p>3. Kurose, J.F., Ross, K.W., Computer Networking, 8th ed., Pearson, 2021.</p> <p>Bibliografie complementara laborator</p> <p>4. Mosquitto MQTT Documentation: https://mosquitto.org/documentation/</p> <p>5. gRPC Documentation: https://grpc.io/docs/</p> <p>Resurse online gratuite recomandate</p> <p>6. Wireshark Documentation: https://www.wireshark.org/docs/</p> <p>7. GNS3 Documentation: https://docs.gns3.com/</p> <p>8. gRPC Documentation: https://grpc.io/docs/</p> <p>9. Mosquitto MQTT: https://mosquitto.org/documentation/</p>		

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

Disciplina acopera protocoalele utilizate in sisteme reale de productie — de la QUIC si gRPC din platformele web, la MQTT si OPC-UA din sistemele industriale. Competenta de a alege si justifica protocolul adecvat unui scenariu tehnic este formata direct prin laboratoare si proiectul integrat. Corelarea cu A11 (proiectarea retelelor), R11 (migrarea in infrastructuri distribuite) si R23 (validarea solutiilor cu beneficiarii) reflecta aplicabilitatea directa in proiectele de integrare a sistemelor.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoasterea protocoalelor avansate de transport, rutare, securitate si aplicatii distribuite; justificarea alegerii unui protocol pentru un scenariu dat.	Examen scris: analiza unui protocol dat, configurarea unui scenariu de rutare si justificarea alegerii unui protocol	40%
8.5 Seminar/laborator	Corectitudinea implementarilor din laborator si calitatea justificarii tehnice din proiectul integrat de la L7.	Implementari predate pe Sakai dupa fiecare lucrare (50%) si proiect integrat cu colocviu oral la L7 (50%)	60%
8.6 Standard minim de performanță:			
Justificarea la colocviu a alegerii protocolului din proiectul integrat cu cel puțin doua argumente tehnice cuantificabile (latenta, overhead, garantii de livrare) fata de o alternativa concreta.			

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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei	INTELIGENȚĂ ARTIFICIALĂ – TIDS402						
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	IV	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	Proiectarea Algoritmilor (TIDS307); Matematici Aplicare (TIDS206) — algebra liniara, probabilitati; Baze de Date (TIDS305).
3.2 de competențe	Cunoasterea algebrei liniare (vectori, matrice, gradient) si a probabilitatilor de baza; programare Python cu NumPy la nivel mediu; cunoasterea structurilor de date (arbori, grafuri).

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sala de curs cu videoproiector si acces la internet; platforma Sakai; recomandare: Python cu NumPy, scikit-learn, PyTorch si Jupyter instalate inainte de curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator cu calculatoare individuale sau acces la Google Colab (GPU gratuit); Python cu scikit-learn, PyTorch, Matplotlib; studentii 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	C17. Cunoaște în profunzime principiile inteligenței artificiale — algoritmi de căutare, reprezentarea cunoștințelor, sisteme expert — și metodele de învățare automată (Machine Learning), cu conștientizarea critică a limitelor și implicațiilor etice ale acestora.
5.2 Aptitudini	A17. Proiectează și implementează sisteme de inteligență artificială și modele de învățare automată (Machine Learning), evaluând critic performanța acestora și adaptând hiperparametrii pentru optimizarea rezultatelor.
5.3 Responsabilitate și autonomie	R17. Acționează în mod responsabil și etic în toate activitățile profesionale din domeniul TI, respectând drepturile de proprietate intelectuală, legislația privind protecția datelor cu caracter personal (GDPR) și standardele de conduită profesională.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Introducere in inteligenta artificiala. Definitia si ramurile IA: simbolica vs. statistica. Testul Turing, agentii rationali. Aplicatii actuale: recomandare, recunoastere vocala, conducere autonoma.	2	Prelegere interactiva	
Tema 2: Cautare neinformata. Formularea problemelor ca spatii de stari. BFS, DFS, IDDFS: completitudine, optimalitate, complexitate. Problema puzzle-ului de 8.	2	Prelegere interactiva	
Tema 3: Cautare informata. Functia euristica: admisibilitate, consistenta. Algoritmul A*: demonstrarea optimalitatii. IDA*. Euristica Manhattan si Hamming.	2	Prelegere interactiva	
Tema 4: Cautare in jocuri adversariale. Jocuri cu suma zero, arborele de joc. Minimax si taietura alfa-beta. Monte Carlo Tree Search (MCTS).	2	Prelegere interactiva	
Tema 5: Satisfacerea constrangerilor (CSP). Backtracking cu forward checking, propagarea constrangerilor AC-3. Euristica MRV. Aplicatii: sudoku, colorarea grafurilor.	2	Prelegere interactiva	
Tema 6: Reprezentarea cunostintelor si logica. Logica de ordinul I, rezolutia. Sisteme expert: forward si backward chaining. Ontologii si OWL.	2	Prelegere interactiva	
Tema 7: Fundamente ale Machine Learning. Bias-variance tradeoff, overfitting, regularizare L1/L2. Validare incrucisata. Metrici: accuracy, F1, ROC-AUC.	2	Prelegere interactiva	
Tema 8: Clasificare si regresie. Regresia liniara si logistica, gradientul descendent. k-NN, Decision Trees. Random Forest: bagging, importanta caracteristicilor.	2	Prelegere interactiva	
Tema 9: SVM si metode de ansamblu. SVM: kerneluri RBF si polinomial. AdaBoost, Gradient Boosting, XGBoost. Compararea metodelor de ansamblu.	2	Prelegere interactiva	
Tema 10: Retele neuronale artificiale. Perceptronul, retea feedforward,	2	Prelegere interactiva	

backpropagation. Functii de activare: ReLU, GELU. Batch normalization, dropout.			
Tema 11: Deep Learning — CNN si RNN. Retele convolutive: strat convolutional, pooling, ResNet. Transfer learning. LSTM si GRU: problema gradientului disparut.	2	Prelegere interactiva	
Tema 12: Procesarea limbajului natural. TF-IDF, Word2Vec. Arhitectura Transformer: self-attention, BERT, GPT. Aplicatii: NER, traducere automata.	2	Prelegere interactiva	
Tema 13: Reinforcement Learning. Procese Markov de decizie, Q-Learning, ecuatia Bellman. Deep Q-Network (DQN). Aplicatii: jocuri, robotica.	2	Prelegere interactiva	
Tema 14: Etica in IA si recapitulare. Biasul algoritmic, explicabilitatea (LIME, SHAP). Regulamentul AI Act. Principii AI responsabil. Recapitulare generala.	2	Prelegere, recapitulare	
	<p>Bibliografie obligatoric</p> <ol style="list-style-type: none"> 1. Russell, S., Norvig, P., Artificial Intelligence: A Modern Approach, 4th ed., Pearson, 2020. 2. Goodfellow, I., Bengio, Y., Courville, A., Deep Learning, MIT Press, 2016. Disponibil gratuit la: https://www.deeplearningbook.org/ 3. Geron, A., Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 3rd ed., O'Reilly Media, 2022. <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Prince, S.J.D., Understanding Deep Learning, MIT Press, 2023. Disponibil gratuit la: https://udlbook.github.io/udlbook/ 5. Sutton, R.S., Barto, A.G., Reinforcement Learning: An Introduction, 2nd ed., MIT Press, 2018. Disponibil gratuit la: http://incompleteideas.net/book/the-book-2nd.html <p>Resurse software:</p> <ol style="list-style-type: none"> 6. scikit-learn Documentation: https://scikit-learn.org/stable/ 7. PyTorch Documentation: https://pytorch.org/docs/ 8. Google Colab: https://colab.research.google.com/ 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Algoritmi de cautare. Implementarea BFS, DFS si A* pentru puzzle-ul de 8 in Python. Compararea nodurilor expandate si a timpului. Testarea cu euristica Manhattan.	2	Lucrare practica Python	
Lucrarea L2: Clasificare cu scikit-learn. Antrenarea a trei clasificatori (Logistic Regression, Decision Tree, Random Forest) pe un dataset real. Compararea metricilor accuracy, F1, ROC-AUC.	2	Lucrare practica Python, scikit-learn	
Lucrarea L3: Preprocesare si selectia caracteristicilor. Tratarea valorilor lipsa, normalizarea, encoding pe dataset Titanic. Aplicarea PCA. Compararea performantei cu si fara preprocesare.	2	Lucrare practica Python, scikit-learn	
Lucrarea L4: Retea neuronală cu PyTorch. Implementarea unui MLP pentru clasificare binară cu Adam si BCELoss. Trasarea curbilor de learning. Experimentarea cu dropout.	2	Lucrare practica Python, PyTorch	
Lucrarea L5: Transfer learning cu CNN. Fine-tuning al unui model ResNet-18 pretrained pe un dataset de imagini (Google Colab). Compararea cu antrenarea de la zero.	2	Lucrare practica PyTorch, Colab	

Lucrarea L6: NLP — clasificarea textului. Clasificator de sentiment pe IMDB cu TF-IDF + Logistic Regression si cu BERT pretrained (Hugging Face). Compararea celor doua abordari.	2	Lucrare practica Python, HuggingFace	
Lucrarea L7: Proiect integrat si colocviu. Pipeline ML complet: problema definita, preprocesare, cel putin 2 modele comparate cu metrici, analiza erorilor si a biasului. Colocviu oral.	2	Colocviu	
<p>Bibliografie obligatorie laborator</p> <ol style="list-style-type: none"> 1. scikit-learn Documentation: https://scikit-learn.org/stable/ 2. PyTorch Documentation: https://pytorch.org/docs/ 3. Geron, A., Hands-On Machine Learning, 3rd ed., O'Reilly Media, 2022. <p>Bibliografie complementara laborator</p> <ol style="list-style-type: none"> 4. Hugging Face Documentation: https://huggingface.co/docs 5. Kaggle Datasets si tutoriale: https://www.kaggle.com/ <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. scikit-learn Documentation: https://scikit-learn.org/stable/ 7. PyTorch Documentation: https://pytorch.org/docs/ 8. Hugging Face Documentation: https://huggingface.co/docs 9. Google Colab: https://colab.research.google.com/ 			

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

<p>Disciplina acopera spectrul complet al IA: algoritmi clasici de cautare, ML supervizat, deep learning si reinforcement learning. Pipeline-ul ML complet din laborator formeaza direct A17.</p> <p>Tema 14 si analiza biasului din proiectul integrat formeaza R17: responsabilitatea etica fata de datele de antrenament, GDPR si impactul social al sistemelor IA.</p>
--

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoasterea algoritmilor de cautare (A*, minimax) si a metodelor ML (clasificare, deep learning, RL); alegerea si justificarea unui model pentru un scenariu dat; identificarea surselor de bias si a implicatiilor etice.	Examen scris: problema de cautare (A* cu euristica data), analiza unui model ML si identificarea unui risc etic intr-un sistem IA descris	30%
8.5 Seminar/laborator	Corectitudinea implementarilor din laborator si calitatea analizei comparative din proiectul integrat (cel putin 2 modele comparate cu metrici si analiza biasului).	Implementari predate pe Sakai dupa fiecare lucrare (50%) si proiect integrat cu colocviu oral la L7 (50%)	70%
8.6 Standard minim de performanță:			

Prezentarea la colocviu a cel puțin doua modele comparate pe același set de date cu metrice calculate corect și a unei secțiuni de analiză a biasului. Un proiect cu un singur model sau fără analiză erorilor nu îndeplinește standardul minim.

Data completării 16.03.2026	Semnătura titularului de curs Lect. Dr. Burlacu Cătălina- Mercedes	Semnătura titularului de seminar / laborator Lect. Dr. Burlacu Cătălina- Mercedes
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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei				ARHITECTURI PENTRU PRELUCRARE NUMERICĂ DE SEMNAL- PROIECT– TIDS402			
1.2 Titularul activităților de curs							
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator / proiect				Conf. Dr. Lacrămă Dan-Laurențiu			
1.5 Anul de studiu	IV	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					11
Documentare					4
Studiu individual					3
Referate					
Teme casă					
Proiect					2
Alte activități (inclusive colocviu, examen)					2
2.9 Total ore pe semestru					25
2.10 Numărul de credite					1

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Arhitecturi pentru Prelucrare Numerica de Semnal (TIDS311) — disciplina de baza corespunzatoare acestui proiect.
3.2 de competențe	Parcurgerea sau parcurgerea concomitenta a disciplinei TIDS311; capacitatea de a implementa filtre FIR/IIR si FFT in Python sau MATLAB; familiarizarea cu formatele in virgula fixa si cu arhitectura DSP.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	
4.2 de desfășurare a seminarului/laboratorului	Activitatea de proiect se desfasoara individual, la calculatorul propriu sau in laborator; Python cu NumPy, SciPy, fixedpoint/fxpmath si MATLAB/Simulink; intalnirile saptamanale de indrumare au loc fata in fata sau online; predarea 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	<p>C4. Deține cunoștințe specializate privind arhitecturile sistemelor de calcul, organizarea procesoarelor, ierarhia memoriei și interfețele hardware-software, inclusiv tendințele actuale în proiectarea procesoarelor de semnal digital (DSP).</p> <p>C5. Cunoaște în profunzime principiile proiectării logice și ale electronicii digitale, inclusiv arhitecturile pentru prelucrare numerică de semnal, situate la avangarda domeniului.</p>
5.2 Aptitudini	A4. Elaborează documentație tehnică completă și clară — specificații, diagrame de flux, manuale de utilizare — în conformitate cu standardele ingineresti și cerințele legale aplicabile.
5.3 Responsabilitate și autonomie	R4. Acționează autonom în selectarea și utilizarea tehnologiilor, framework-urilor și instrumentelor software adecvate contextului tehnic, justificând critic alegerile făcute în fața echipei și a beneficiarilor.

6. Conținuturi

6.1 Proiect	Nr. ore	Metode de predare	Observații
P1: Alegerea temei și definirea cerintelor. Prezentarea categoriilor de proiecte: implementarea unui filtru DSP pe arhitectura tinta, proiectarea unui sistem de achiziție și procesare, compararea arhitecturilor DSP vs. FPGA. Alegerea temei și redactarea documentului de cerințe: specificațiile sistemului, metricile de performanță a tinta.	1	Indrumare individuala	
P2: Analiza algoritmului și alegerea arhitecturii. Identificarea algoritmului DSP de implementat (FIR, IIR, FFT, modulație/demodulație). Calculul cerintelor de calcul: numărul de multiplicații și adunări pe esanțion. Alegerea arhitecturii tinta (DSP software, FPGA HLS, GPU/Numba) cu justificare tehnică cantitativă.	1	Indrumare, feedback	
P3: Implementarea în virgulă mobilă. Implementarea de referință a algoritmului în Python cu NumPy/SciPy în virgulă mobilă (float64). Validarea funcționalității: compararea cu rezultatele teoretice. Profilarea execuției: identificarea secțiunilor computaționale dominante.	1	Implementare, indrumare	
P4: Analiza efectelor virgulei fixe. Conversia algoritmului la reprezentarea în virgulă fixă: calculul formatului Q necesar pentru coeficienți și date. Simularea efectelor de cuantizare cu fpxmath. Analiza degradării răspunsului în frecvență și a riscului de overflow.	1	Implementare, indrumare	
P5: Optimizarea implementării. Aplicarea a cel puțin unei optimizări specifice arhitecturii alese: vectorizarea SIMD (Numba), loop unrolling, implementarea polyphase (pentru filtre multirate) sau pipeline HLS. Măsurarea impactului optimizării asupra vitezei de execuție.	1	Implementare, profilare	

P6: Simularea pe arhitectura tinta — faza 1. Prima rulare completa a implementarii pe arhitectura tinta (DSP simulat, FPGA HLS sau GPU). Masurarea timpului de executie si a consumului de resurse. Identificarea discrepantelor fata de specificatii.	1	Implementare, masuratori	
P7: Simularea pe arhitectura tinta — faza 2. Remedierea problemelor identificate la P6. A doua rundă de masuratori: latentă, throughput, consumul de resurse (LUT/DSP48 pentru FPGA sau cicluri per esantion pentru DSP). Compararea cu specificatiile initiale.	1	Implementare, masuratori	
P8: Prezentare intermediara. Prezentarea stadiului proiectului: cerintele, arhitectura aleasa cu justificarea cantitativa, implementarea in virgula fixa si rezultatele masurate. Feedback din partea titularului si ajustarea directiei pentru a doua jumătate.	1	Prezentare intermediara	
P9: Testarea robustetii. Testarea implementarii pe semnale de intrare variate: semnal sinusoidal, semnal compus, zgomot alb, semnal impulsiv. Verificarea stabilitatii filtrului IIR (daca este cazul). Documentarea cazurilor limita si a comportamentului la overflow.	1	Testare, documentare	
P10: Comparatia arhitecturala. Implementarea sau estimarea performantei pentru o a doua arhitectura alternativa (ex: daca proiectul principal este pe DSP, estimarea pe FPGA si invers). Comparatia cantitativa: latentă, throughput, putere consumata, costul implementarii.	1	Analiza comparativa	
P11: Redactarea documentatiei tehnice. Structura documentatiei: specificatiile sistemului, alegerea arhitecturii cu justificarea cantitativa, descrierea implementarii cu diagrame de flux, rezultatele masurate, comparatia cu specificatiile, comparatia arhitecturala.	1	Redactare tehnica	
P12: Revizuirea documentatiei. Revizuirea documentatiei pe baza feedback-ului de la P8 si P11. Curatarea codului sursa: comentarii, README cu instructiuni de rulare. Pregatirea demonstratiei live pentru prezentarea finala.	1	Revizuire, finalizare	
P13: Repetitia prezentarii finale. Prezentarea completa a proiectului in fata titularului in format de repetitie: 10 minute + feedback. Ajustarea prezentarii si a demonstratiei pe baza observatiilor primite.	1	Repetitie prezentare	
P14: Prezentarea finala si evaluare. Prezentare orala (10 minute + 5 minute Q&A): cerintele, arhitectura aleasa cu argumentele cantitative, demonstratie live a rularii pe cel putin un semnal de test, curbele de performanta, comparatia arhitecturala. Predarea documentatiei si a codului pe Sakai.	1	Prezentare orala, evaluare	

	<p>Bibliografie obligatorie proiect</p> <ol style="list-style-type: none"> 1. Proakis, J.G., Manolakis, D.G., Digital Signal Processing, 4th ed., Pearson, 2006. 2. SciPy Signal Processing Documentation: https://docs.scipy.org/doc/scipy/reference/signal.html 3. DSPGuide — carte gratuita DSP: https://www.dspguide.com/ <p>Bibliografie complementara proiect</p> <ol style="list-style-type: none"> 4. Oppenheim, A.V., Schaffer, R.W., Discrete-Time Signal Processing, 3rd ed., Pearson, 2009. 5. Vitis HLS User Guide (UG1399): https://docs.xilinx.com/r/en-US/ug1399-vitis-hls <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. SciPy Signal Processing: https://docs.scipy.org/doc/scipy/reference/signal.html 7. DSPGuide: https://www.dspguide.com/ 8. Numba CUDA Documentation: https://numba.readthedocs.io/en/stable/cuda/ 9. Vitis HLS User Guide: https://docs.xilinx.com/r/en-US/ug1399-vitis-hls
--	---

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

Proiectul integreaza competentele din TIDS311 intr-un ciclu complet: analiza algoritmului, implementarea in virgula fixa, optimizarea pentru arhitectura tinta si documentatia tehnica. Ciclul reproduce contextul unui proiect de optimizare DSP din industrie.

Justificarea cantitativa a alegerii arhitecturii (C4, C5, R4) si documentatia tehnica conform standardelor ingineresti (A4) sunt evaluate explicit la prezentarea finala.

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	Functionarea corecta a implementarii (rezultatele virgula fixa concordante cu referinta virgula mobila); calitatea documentatiei tehnice (specificatii, diagrame, justificarea cantitativa a alegerii arhitecturii, curbele de performanta masurate); calitatea prezentarii orale.	Documentatie si cod predate pe Sakai (50%) si prezentare orala finala la P14 cu demonstratie live (50%)	100%
8.6 Standard minim de performanță:			
Demonstrarea la P14 a implementarii functionale in virgula fixa pe cel putin un semnal de test si prezentarea justificarii cantitative a alegerii arhitecturii (cel putin doua argumente numerice: latentă sau throughput si consumul de resurse). O implementare fara analiza virgulei fixe sau o justificare arhitecturala fara argumente cantitative nu indeplinesc standardul minim.			

Data completării 19.03.2026	Semnătura titularului de curs	Semnătura titularului de seminar / laborator/proiect 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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei		SECURITATEA INFORMAȚIEI ȘI A COMUNICAȚIILOR – TIDS404					
1.2 Titularul activităților de curs		Conf. Dr. Postolache Florin					
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator		Conf. Dr. Postolache Florin					
1.5 Anul de studiu	IV	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					69
Documentare					22
Studiu individual					21
Referate					20
Teme casă					3
Proiect					
Alte activități (inclusive colocviu, examen)					3
2.9 Total ore pe semestru					125
2.10 Numărul de credite					5

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Retele de Calculatoare (TIDS301); Protocoale de Comunicatii (TIDS401); Sisteme de Operare (TIDS302).
3.2 de competențe	Cunoasterea stivei TCP/IP si a modelului OSI; familiarizarea cu linia de comanda Linux; notiuni de baza de criptografie (algoritmi simetrici si asimetrici).

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sala de curs cu videoproiector si acces la internet; platforma Sakai.
4.2 de desfășurare a seminarului/laboratorului	Laborator cu calculatoare Linux (sau masini virtuale Kali Linux); Wireshark, Nmap, Metasploit Framework, OpenSSL instalate; acces la un mediu de testare izolat (VirtualBox cu retea host-only).

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

5.1 Cunoștințe	<p>C15. Deține cunoștințe aprofundate privind principiile securității informației și comunicațiilor, standardele internaționale de securitate (ISO 27000), tehnicile de criptografie simetrică și asimetrică și protocoalele de securitate a rețelelor.</p> <p>C16. Posedă cunoștințe specializate privind vulnerabilitățile sistemelor informatice și mecanismele de protecție, cu conștientizarea critică a amenințărilor cibernetice actuale și a cadrului legal în domeniu.</p>
5.2 Aptitudini	A15. Identifică vulnerabilitățile sistemelor informatice și implementează măsuri de securitate adecvate — firewall, criptare, autentificare, control acces — în conformitate cu standardele internaționale de securitate (ISO 27000, GDPR).
5.3 Responsabilitate și autonomie	<p>R12. Acționează autonom în implementarea și menținerea securității sistemelor informatice, asumându-și responsabilitatea pentru identificarea vulnerabilităților, aplicarea corecțiilor și conformitatea cu standardele de securitate în vigoare.</p> <p>R15. Evaluează critic literatura tehnică de specialitate și tendințele emergente din domeniu — inteligență artificială, cloud computing, securitate cibernetică — și integrează în mod responsabil noile cunoștințe în practica profesională proprie și a echipei.</p>

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Principiile securității informației. Triada CIA: confidențialitate, integritate, disponibilitate. Autentificarea, autorizarea, non-repudierea. Modelele de securitate: Bell-LaPadula, Biba. Standardele ISO 27001/27002: domeniile de control, ISMS.	3	Prelegere interactivă	
Tema 2: Criptografie simetrică. Cifrurile bloc: AES (structura round-urilor, modulele ECB/CBC/GCM). Cifrurile flux: ChaCha20. Funcțiile hash: SHA-256, SHA-3. HMAC. Gestionarea cheilor simetrice: Key Derivation Functions (PBKDF2, Argon2).	3	Prelegere interactivă	
Tema 3: Criptografie asimetrică și PKI. RSA: generarea cheilor, criptarea, semnatura digitală. Criptografia pe curbe eliptice (ECDSA, ECDH). Infrastructura PKI: CA, certificate X.509, lanțul de încredere, CRL și OCSP. Certificate transparente (CT logs).	3	Prelegere interactivă	
Tema 4: Securitatea protocoalelor de rețea. TLS 1.3: handshake-ul, suite-le de cifrare, HSTS. SSH: autentificarea cu chei publice, tunelarea. VPN: IPsec și WireGuard. DNS Security: DNSSEC, DoH, DoT. Atacuri: SSL stripping, downgrade, MITM.	3	Prelegere interactivă	
Tema 5: Vulnerabilități și atacuri aplicative. OWASP Top 10 (2021): SQL Injection, XSS, CSRF, IDOR, SSRF, deserializare nesigură. Atacuri de tip injection: mecanismele, exploatarea, mitigarea. Buffer overflow: stack smashing, protecții ASLR/DEP/canary.	3	Prelegere interactivă	

Tema 6: Securitatea sistemelor de operare. Hardening Linux: reducerea suprafetei de atac, SELinux/AppArmor, auditd. Gestionarea privilegiilor: sudo, capabilities, chroot, namespaces, cgroups. Windows security: Active Directory, GPO, Credential Guard, BitLocker.	3	Prelegere interactiva	
Tema 7: Securitatea retelelor. Firewall-uri: packet filtering, stateful inspection, next-gen (NGFW). IDS/IPS: Snort, Suricata, regulile de detectie. Network segmentation: DMZ, VLAN, zero-trust networking. Honeypots: tipuri, scopul, implementarea.	3	Prelegere interactiva	
Tema 8: Autentificarea si managementul identitatii. Autentificarea multi-factor (MFA): TOTP (RFC 6238), FIDO2/WebAuthn, passkeys. SSO: SAML 2.0, OAuth 2.0, OpenID Connect. Privileged Access Management (PAM). Atacuri: credential stuffing, phishing, SIM swapping.	3	Prelegere interactiva	
Tema 9: Testarea penetrare (penetration testing). Metodologia: reconnaissance, scanning, exploitation, post-exploitation, reporting. Instrumente: Nmap, Metasploit, Burp Suite, Hydra, John the Ripper. CVE si CVSS: scoring-ul vulnerabilitatilor. Raportul de pentest: structura, clasificarea riscurilor.	3	Prelegere interactiva	
Tema 10: Securitatea aplicatiilor web si API. Secure development lifecycle (SDL). SAST si DAST: SonarQube, OWASP ZAP. Securitatea API REST: autentificarea, rate limiting, validarea input-ului. Content Security Policy (CSP), SameSite cookies. WAF: regulile de protectie.	3	Prelegere interactiva	
Tema 11: Securitatea in cloud si containere. Modelul shared responsibility in cloud. IAM in AWS/Azure: principiul least privilege. Securitatea containerelor: imaginile signed, Trivy scanning, Falco runtime security, Kubernetes Network Policies. Secretele in cloud: Vault, AWS Secrets Manager.	3	Prelegere interactiva	
Tema 12: Managementul incidentelor si raspunsul la incidente. NIST Incident Response Framework: preparare, detectie, containment, eradicare, recuperare, lectii invatate. SIEM: Splunk, Elastic SIEM — colectarea logurilor, regulile de detectie. Forensics digital: colectarea probelor, lantul de custodie.	3	Prelegere interactiva	
Tema 13: Cadrul legal si conformitatea. GDPR: principiile, drepturile subiectilor, obligatiile operatorului, notificarea bresei. Directiva NIS2: sectoarele critice, cerintele de securitate. AI Act: implicatiile pentru sistemele AI cu risc ridicat. ISO 27001: procesul de certificare, auditul intern.	3	Prelegere interactiva	

<p>Tema 14: Amenintari avansate si recapitulare. APT (Advanced Persistent Threats): ciclul de viata, tactici MITRE ATT&CK. Ransomware: mecanismele, prevenirea, recuperarea. Supply chain attacks: SolarWinds, XZ Utils. Securitatea IoT: constrangerile, atacurile, standardele (ETSI EN 303 645). Recapitulare generala.</p>	3	Prelegere, recapitulare	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> 1. Stallings, W., Brown, L., Computer Security: Principles and Practice, 4th ed., Pearson, 2018. 2. Anderson, R., Security Engineering, 3rd ed., Wiley, 2020. Disponibil gratuit la: https://www.cl.cam.ac.uk/~rja14/book.html 3. OWASP Top 10 (2021): https://owasp.org/www-project-top-ten/ <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Shostack, A., Threat Modeling: Designing for Security, Wiley, 2014. 5. NIST Cybersecurity Framework 2.0: https://www.nist.gov/cyberframework <p>Resurse software:</p> <ol style="list-style-type: none"> 6. OWASP Documentation: https://owasp.org/ 7. NIST Cybersecurity Framework: https://www.nist.gov/cyberframework 8. MITRE ATT&CK Framework: https://attack.mitre.org/ 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
<p>Lucrarea L1: Criptografie cu OpenSSL. Criptarea si decriptarea unui fisier cu AES-256-GCM. Generarea unei perechi de chei RSA-4096 si semnarea unui document. Verificarea unui certificat X.509: lantul de incredere, datele de expirare, extensiile.</p>	2	Lucrare practica Linux, OpenSSL	
<p>Lucrarea L2: Analiza traficului si atacuri de retea. Capturarea traficului HTTP si HTTPS cu Wireshark: compararea vizibilitatii datelor. Simularea unui atac ARP spoofing intr-o retea host-only VirtualBox. Detectarea atacului in Wireshark si configurarea unui ARP static pentru mitigare.</p>	2	Lucrare practica Wireshark, Linux	
<p>Lucrarea L3: Scanare si enumerare cu Nmap. Scanarea unui host tinta (masina virtuala vulnerabila intentionat — Metasploitable sau DVWA). Identificarea porturilor deschise, serviciilor si versiunilor. Detectarea OS-ului. Exportul rezultatelor in format XML si interpretarea lor.</p>	2	Lucrare practica Nmap, Kali Linux	
<p>Lucrarea L4: Exploatarea vulnerabilitatilor cu Metasploit. Exploatarea unei vulnerabilitati cunoscute pe masina virtuala tinta (mediu izolat). Obtinerea unui shell Meterpreter. Post-exploitation: enumerarea utilizatorilor, dump credentiale. Redactarea unui raport de vulnerabilitate cu CVSS score.</p>	2	Lucrare practica Metasploit, mediu izolat	
<p>Lucrarea L5: Securitatea aplicatiilor web — OWASP. Exploatarea SQL Injection si XSS pe DVWA (Damn Vulnerable Web App). Utilizarea Burp Suite pentru interceptarea si modificarea cererilor. Configurarea unui Content Security Policy si verificarea eficacitatii impotriva XSS.</p>	2	Lucrare practica Burp Suite, DVWA	
<p>Lucrarea L6: Hardening Linux si IDS. Aplicarea unui checklist de hardening pe o</p>	2	Lucrare practica Linux, Snort	

masina virtuala Ubuntu: dezactivarea serviciilor inutile, configurarea ufw, SSH cu chei publice, fail2ban. Configurarea Snort cu o regula personalizata si testarea detectiei.			
Lucrarea L7: Proiect integrat si colocviu. Realizarea unui audit de securitate complet pe un sistem tinta desemnat: scanare, identificarea vulnerabilitatilor, exploatarea controlata a cel putin uneia, propunerea masurilor de remediere. Raport de pentest cu clasificarea riscurilor dupa CVSS. Colocviu oral.	2	Colocviu	
	<p>Bibliografie obligatorie laborator</p> <p>1. OWASP Testing Guide v4.2: https://owasp.org/www-project-web-security-testing-guide/</p> <p>2. Metasploit Documentation: https://docs.metasploit.com/</p> <p>3. Nmap Documentation: https://nmap.org/docs.html</p> <p>Bibliografie complementara laborator</p> <p>4. Kali Linux Documentation: https://www.kali.org/docs/</p> <p>5. Anderson, R., Security Engineering, 3rd ed., Wiley, 2020. Disponibil gratuit la: https://www.cl.cam.ac.uk/~rja14/book.html</p> <p>Resurse online gratuite recomandate</p> <p>6. OWASP Testing Guide: https://owasp.org/www-project-web-security-testing-guide/</p> <p>7. Metasploit Documentation: https://docs.metasploit.com/</p> <p>8. MITRE ATT&CK: https://attack.mitre.org/</p> <p>9. HackTheBox — platforma de practica CTF: https://www.hackthebox.com/</p>		

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

<p>Disciplina acopera securitatea de la nivel criptografic pana la testarea penetrare si managementul incidentelor. Abordarea ofensiva din laborator (scanare, exploatare controlata, raport de pentest) formeaza direct A15 si R12 prin expunerea practica la vulnerabilitatile reale si la responsabilitatea remediei lor.</p> <p>Cadrul legal (NIS2, GDPR, AI Act) si evaluarea critica a tendintelor emergente din Tema 14 formeaza R15: capacitatea de a integra responsabil noile cunostinte de securitate in practica profesionala.</p>
--

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoasterea principiilor CIA, standardelor ISO 27001/27002, criptografiei simetrice si asimetrice, protocoalelor TLS/SSH/IPsec; cunoasterea categoriilor OWASP Top 10 si a metodologiei de pentest; identificarea implicatiilor legale GDPR si NIS2 pentru un scenariu dat.	Examen scris: analiza unui scenariu de atac (identificarea vectorului, a vulnerabilitatii si a masurilor de remediere), o problema de criptografie aplicata si o intrebare despre conformitatea legala	40%
8.5 Seminar/laborator	Calitatea raportului de pentest de la L7: identificarea corecta a	Implementari predate pe Sakai dupa fiecare lucrare (50%) si	60%

	vulnerabilitatilor, clasificarea CVSS si propunerea masurilor de remediere; corectitudinea implementarilor din laboratoare (L1-L6).	raport de pentest cu colocviu oral la L7 (50%)	
8.6 Standard minim de performanță:			
Predarea unui raport de pentest care include cel puțin o vulnerabilitate exploatarea demonstrabil (cu screenshot), scorul CVSS calculat corect si masurile de remediere concrete. Un raport fara exploatarea demonstrata sau fara CVSS score nu indeplineste standardul minim.			

Data completării	Semnătura titularului de curs Conf. Dr. Postolache Florin	Semnătura titularului de seminar / laborator Conf. Dr. Postolache Florin
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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei		TEHNOLOGII DE PROGRAMARE ÎN INTERNET – TIDS405					
1.2 Titularul activităților de curs		Conf. Dr. Bucea Manea Radu Țonis					
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator		Conf. Dr. Bucea Manea Radu Țonis					
1.5 Anul de studiu	IV	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					69
Documentare					22
Studiu individual					22
Referate					21
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					4
2.9 Total ore pe semestru					125
2.10 Numărul de credite					5

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Ingineria Programelor (TIDS308); Baze de Date (TIDS305); Rețele de Calculatoare (TIDS301).
3.2 de competențe	Cunoașterea HTML/CSS/JavaScript la nivel de baza; familiarizarea cu HTTP si REST; experienta cu Git si cu cel puțin un limbaj de programare server-side (Python, Java sau Node.js).

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sala de curs cu videoproiector si acces la internet; platforma Sakai; recomandare: Node.js, VS Code si extensia REST Client instalate inainte de curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator cu calculatoare individuale; Node.js, VS Code, Docker Desktop, Postman; acces la GitHub; studentii 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	C20. Cunoaște în profunzime tehnologiile de programare în Internet — arhitecturi web, servicii REST/SOAP, framework-uri moderne — și principiile dezvoltării aplicațiilor mobile, cu conștientizarea critică a standardelor W3C și a tendințelor actuale.
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. A20. Dezvoltă aplicații web și mobile utilizând tehnologii și framework-uri moderne, respectând standardele W3C, principiile de securitate și cerințele de performanță și accesibilitate ale produselor software.
5.3 Responsabilitate și autonomie	R20. Gestionează autonom proiecte de inginerie software de complexitate ridicată, planificând activitățile, estimând și alocând resursele tehnice și umane necesare și adaptând planul de proiect la schimbările imprevizibile apărute pe parcurs.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Arhitecturi web moderne. Evoluția web: static, dinamic, SPA, SSR, SSG. Arhitectura client-server și microservicii. Protocolul HTTP/2 și HTTP/3: multiplexarea, push, prioritizarea. WebSockets: comunicarea full-duplex. Comparatia arhitecturilor pentru scenarii reale.	2	Prelegere interactiva	
Tema 2: HTML5 și CSS3 avansat. Elementele semantice HTML5: article, section, aside, main. CSS Grid și Flexbox: modelele de layout, comparatia. CSS Custom Properties (variabile). Animatii CSS: transitions, keyframes, performance. Atributele ARIA și accesibilitatea WCAG 2.1.	2	Prelegere interactiva	
Tema 3: JavaScript modern (ES2020+). Promises, async/await, event loop. Modulele ES: import/export, dynamic import. Destructuring, spread/rest, optional chaining. Proxy și Reflect. Web APIs: Fetch, Intersection Observer, Web Workers.	2	Prelegere interactiva	
Tema 4: Framework-uri JavaScript frontend. React: componentele functionale, hooks (useState, useEffect, useContext, useReducer), Virtual DOM, reconcilierea. Vue 3: Composition API. Angular: arhitectura, dependency injection. Comparatia framework-urilor: ecosistem, performanta, curba de invatare.	2	Prelegere interactiva	
Tema 5: Managementul starii in aplicatii web. Problema starii globale. Redux: store, actions, reducers, middleware (Redux Thunk). Zustand și Jotai: alternative lightweight. React Query: fetching și caching serverului. Context API vs. state management extern.	2	Prelegere interactiva	
Tema 6: Server-Side Rendering și Static Site Generation. Next.js: SSR	2	Prelegere interactiva	

(getServerSideProps), SSG (getStaticProps), ISR (Incremental Static Regeneration), App Router. Nuxt.js pentru Vue. Avantajele pentru SEO si performanta: Core Web Vitals (LCP, FID, CLS). Comparatie SPA vs. SSR vs. SSG.			
Tema 7: Node.js si Express. Event loop Node.js: call stack, callback queue, microtask queue. Express.js: routing, middleware-urile, gestionarea erorilor. Body parsers, CORS, helmet. Structura unui proiect Express scalabil: MVC, repository pattern.	2	Prelegere interactiva	
Tema 8: API REST avansat si GraphQL. REST: resurse, metode HTTP, coduri de stare, HATEOAS. Versionarea API: URL vs. header. Documentarea cu OpenAPI/Swagger. GraphQL: schema, queries, mutations, subscriptions, N+1 problem. Comparatie REST vs. GraphQL vs. tRPC.	2	Prelegere interactiva	
Tema 9: Autentificare si autorizare web. Session-based vs. token-based authentication. JWT: structura, semnarea, validarea, refresh tokens, vulnerabilitati (alg:none). OAuth 2.0: fluxurile (authorization code, PKCE, client credentials). OpenID Connect. Implementarea in Express cu Passport.js.	2	Prelegere interactiva	
Tema 10: Baze de date pentru aplicatii web. ORM cu Prisma: schema, migratii, queryurile type-safe. MongoDB cu Mongoose: schema flexibila, indexuri, agregari. Redis: caching, session store, pub/sub. Alegerea bazei de date: SQL vs. NoSQL pentru scenarii web.	2	Prelegere interactiva	
Tema 11: Performanta si optimizare web. Code splitting si lazy loading. Bundle optimization: tree shaking, minificare, compresie Brotli. Strategii de caching: HTTP cache headers, Service Workers, Cache API. CDN: functionarea, configurarea. Lighthouse: auditarea si interpretarea scorurilor.	2	Prelegere interactiva	
Tema 12: Testarea aplicatiilor web. Unit testing cu Jest si Vitest. Testing Library: testarea componentelor React din perspectiva utilizatorului. Testarea API cu Supertest. E2E cu Playwright: scenarii, selectori, network mocking. Testarea accesibilitatii: axe-playwright.	2	Prelegere interactiva	
Tema 13: Deployment si CI/CD pentru aplicatii web. Docker pentru aplicatii web: Dockerfile multi-stage, docker-compose. Vercel si Netlify: deployment continuu, preview deployments, edge functions. GitHub Actions: pipeline build-test-deploy. Environment variables si secrets management.	2	Prelegere interactiva	
Tema 14: WebAssembly, PWA si recapitulare. Progressive Web Apps: Service Workers, Web App Manifest, offline-first. WebAssembly: compilarea C/C++/Rust in	2	Prelegere, recapitulare	

browser, cazuri de utilizare. Web Components: Custom Elements, Shadow DOM. Tendinte: server components, edge computing. Recapitulare generala.			
	<p>Bibliografie obligatoric</p> <ol style="list-style-type: none"> 1. Flanagan, D., JavaScript: The Definitive Guide, 7th ed., O'Reilly Media, 2020. 2. Brown, E., Web Development with Node and Express, 2nd ed., O'Reilly Media, 2019. 3. MDN Web Docs — referinta completa HTML, CSS, JavaScript: https://developer.mozilla.org/ <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Simpson, K., You Don't Know JS Yet (seria), 2nd ed., 2020. Disponibil gratuit la: https://github.com/getify/You-Dont-Know-JS 5. Next.js Documentation: https://nextjs.org/docs <p>Resurse software:</p> <ol style="list-style-type: none"> 6. MDN Web Docs: https://developer.mozilla.org/ 7. React Documentation: https://react.dev/ 8. Next.js Documentation: https://nextjs.org/docs 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: HTML5 semantic si CSS Grid. Construirea unui layout de pagina cu HTML5 semantic si CSS Grid. Implementarea designului responsiv cu media queries si CSS custom properties. Validarea cu W3C Validator.	2	Lucrare practica VS Code	
Lucrarea L2: JavaScript modern — Fetch si async/await. Consumarea unui API public (ex. JSONPlaceholder) cu Fetch si async/await. Tratarea erorilor cu try/catch. Afisarea datelor cu DOM manipulation. Utilizarea Intersection Observer pentru lazy loading.	2	Lucrare practica JavaScript	
Lucrarea L3: React — componentele si hooks. Implementarea unei aplicatii React cu componente functionale: lista cu filtrare si sortare. Utilizarea useState, useEffect si useCallback. Ridicarea starii (lifting state up).	2	Lucrare practica React	
Lucrarea L4: React — Context API si React Query. Implementarea temei (dark/light mode) cu Context API. Fetching si caching datelor de la un API cu React Query: loading states, error handling, refetch on focus.	2	Lucrare practica React	
Lucrarea L5: Next.js — SSR si SSG. Implementarea unui mini-blog cu Next.js: pagina de listare (SSG cu getStaticProps) si pagina de detaliu (SSG cu getStaticPaths). Compararea timpilor de incarcare SSG vs. CSR.	2	Lucrare practica Next.js	
Lucrarea L6: API REST cu Express si Prisma. Implementarea unui API REST CRUD pentru o resursa (ex. products) cu Express.js si Prisma (SQLite). Validarea input-ului cu Zod. Documentarea cu Swagger UI.	2	Lucrare practica Node.js, Express	
Lucrarea L7: Autentificare JWT cu Express. Adaugarea autentificarii JWT la API-ul din L6: inregistrare, login, refresh token. Protejarea rutelor cu middleware. Stocarea parolelor cu bcrypt. Testarea cu Postman.	2	Lucrare practica Node.js, Postman	
Lucrarea L8: GraphQL cu Apollo Server. Implementarea unui server GraphQL pentru	2	Lucrare practica GraphQL	

aceasi resursa din L6: schema, queries, mutations. Utilizarea Apollo Sandbox pentru testare. Compararea N+1 problem cu si fara DataLoader.			
Lucrarea L9: Caching cu Redis. Adaugarea Redis ca layer de caching pentru API-ul din L6: cache-ul raspunsurilor GET, invalidarea la PUT/DELETE. Masurarea latentei cu si fara cache. Implementarea session store cu Redis.	2	Lucrare practica Redis, Node.js	
Lucrarea L10: Testarea cu Jest si Testing Library. Scrierea testelor unit pentru functii utilitare. Testarea componentei React din L3 cu React Testing Library: renderizare, user events, queries. Masurarea acoperirii cu Jest coverage.	2	Lucrare practica Jest	
Lucrarea L11: Testare E2E cu Playwright. Scrierea a 5 scenarii E2E pentru aplicatia din L5: navigarea, completarea unui formular, autentificarea. Utilizarea network mocking pentru testarea erorilor API. Rularea in headless mode.	2	Lucrare practica Playwright	
Lucrarea L12: Optimizare si Lighthouse. Auditarea aplicatiei din L5 cu Lighthouse: identificarea problemelor de performanta, accesibilitate si SEO. Implementarea code splitting cu dynamic import. Adaugarea metadatelor Open Graph si a imaginilor optimizate.	2	Lucrare practica Lighthouse, Next.js	
Lucrarea L13: Containerizare cu Docker. Containerizarea aplicatiei fullstack (frontend Next.js + backend Express + PostgreSQL) cu Dockerfile multi-stage si docker-compose. Configurarea variabilelor de mediu. Deployment pe un VPS sau Render.	2	Lucrare practica Docker	
Lucrarea L14: Proiect integrat si colocviu. Prezentarea aplicatiei web fullstack dezvoltate pe parcursul semestrului: arhitectura, deciziile tehnice, testele implementate, scorurile Lighthouse, pipeline-ul de deployment. Colocviu oral.	2	Colocviu	
	<p>Bibliografie obligatorie laborator</p> <ol style="list-style-type: none"> 1. React Documentation: https://react.dev/ 2. Next.js Documentation: https://nextjs.org/docs 3. MDN Web Docs: https://developer.mozilla.org/ <p>Bibliografie complementara laborator</p> <ol style="list-style-type: none"> 4. Prisma Documentation: https://www.prisma.io/docs 5. Playwright Documentation: https://playwright.dev/docs/intro <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. React Documentation: https://react.dev/ 7. Next.js Documentation: https://nextjs.org/docs 8. Prisma Documentation: https://www.prisma.io/docs 9. Playwright Documentation: https://playwright.dev/docs/intro 		

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

Disciplina formeaza competentele fullstack solicitate explicit de industria software: React/Next.js pe frontend, Node.js/Express cu autentificare JWT si caching Redis pe backend, testare automata si

deployment Docker. Aplicatia construita incremental de-a lungul laboratoarelor reproduce structura unui proiect real de produs software.

Managementul deciziilor tehnice justificate la colocviu (A5, A20) si planificarea autonoma a livrarilor saptamanale (R20) formeaza competentele de autonomie profesionala cerute unui inginer software senior.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoasterea arhitecturilor web (SPA, SSR, SSG) si justificarea alegerii pentru un scenariu dat; cunoasterea REST vs. GraphQL, OAuth 2.0/JWT, Core Web Vitals; identificarea vulnerabilitatilor web (CORS gresit, JWT insecure, XSS) dintr-un cod sursa furnizat.	Examen scris: analiza unei arhitecturi web date, proiectarea unui API REST pentru o cerinta specificata si identificarea a doua probleme de securitate sau performanta dintr-un fragment de cod	40%
8.5 Seminar/laborator	Functionarea aplicatiei fullstack din proiectul integrat (frontend, backend, autentificare, deployment Docker); calitatea testelor (cel putin unit + E2E functionale); scorul Lighthouse minim 80 la Performance si Accessibility.	Implementari predate pe Sakai dupa fiecare lucrare (50%) si aplicatie fullstack cu colocviu oral la L14 (50%)	60%
8.6 Standard minim de performanță:			
Demonstrarea la colocviu a aplicatiei fullstack functionale (frontend, backend si autentificare JWT operationale) cu un scor Lighthouse minim 80 la Performance si Accessibility si cu cel putin un test E2E care ruleaza fara erori. O aplicatie fara autentificare functionala sau fara teste nu indeplineste standardul minim.			

Data completării 17.03.2026	Semnătura titularului de curs Conf. Dr. Bucea Manea Radu Țonis	Semnătura titularului de seminar / laborator Conf. Dr. Bucea Manea Radu Țonis
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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei		CALCULATOARE NUMERICE – TIDS406					
1.2 Titularul activităților de curs		Conf. Dr. Bucea Manea Radu Țonis					
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator		Conf. Dr. Bucea Manea Radu Țonis					
1.5 Anul de studiu	IV	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					8
Documentare					3
Studiu individual					3
Referate					3
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					2
2.9 Total ore pe semestru					50
2.10 Numărul de credite					5

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Arhitectura Sistemelor de Calcul (TIDS214); Electronica Digitala (TIDS209); Sisteme de Operare (TIDS302).
3.2 de competențe	Cunoasterea reprezentării numerelor în binar (complement față de 2, virgula mobilă IEEE 754); familiarizarea cu un limbaj de asamblare sau cu C la nivel de pointeri; cunoasterea organizării de bază a unui procesor (fetch-decode-execute).

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sala de curs cu tabla pentru demonstrații, videoproiector și acces la internet; platforma Sakai; recomandare: QEMU, GEM5 sau simulatorul x86 NASM instalat înainte de curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator cu calculatoare individuale; NASM sau GCC cu -O0/-O2 pentru inspectarea codului de asamblare; Valgrind pentru profilarea memoriei; GEM5 sau QEMU pentru simularea arhitecturală; studenții pot folosi propriul laptop cu Linux sau WSL.

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

5.1 Cunoștințe	C4. Deține cunoștințe specializate privind arhitecturile sistemelor de calcul, organizarea procesoarelor, ierarhia memoriei și interfețele hardware-software, inclusiv tendințele actuale în proiectarea procesoarelor de semnal digital (DSP).
5.2 Aptitudini	A4. Elaborează documentație tehnică completă și clară — specificații, diagrame de flux, manuale de utilizare — în conformitate cu standardele ingineresti și cerințele legale aplicabile.
5.3 Responsabilitate și autonomie	R4. Acționează autonom în selectarea și utilizarea tehnologiilor, framework-urilor și instrumentelor software adecvate contextului tehnic, justificând critic alegerile făcute în fața echipei și a beneficiarilor.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Arhitecturi de procesoare. Clasificarea Flynn: SISD, SIMD, MISD, MIMD. Arhitecturi RISC vs. CISC: caracteristicile, exemple (ARM, x86-64). Microarhitectura unui procesor modern: front-end, back-end, unitati de executie. Tendinte: procesoare heterogene (big.LITTLE), RISC-V.	2	Prelegere interactiva	
Tema 2: Pipeline-ul procesorului. Etapele pipeline-ului clasic de 5 stadii: IF, ID, EX, MEM, WB. Hazarduri: structurale, de date (RAW, WAW, WAR), de control. Tehnici de rezolvare: forwarding, stall, branch prediction (static si dinamic). Superpipeline si superscalar.	2	Prelegere interactiva	
Tema 3: Executia out-of-order si speculativa. Algoritmul Tomasulo: reservation stations, register renaming, Common Data Bus. Reorder Buffer (ROB): in-order commit. Executia speculativa: avantajele si costurile (Spectre, Meltdown). SMT (Hyper-Threading): partajarea resurselor.	2	Prelegere interactiva	
Tema 4: Ierarhia memoriei. Principiul localitatii: temporala si spatiaa. Cache L1/L2/L3: organizarea set-associative, politicile de inlocuire (LRU, PLRU), politicile de scriere (write-through, write-back). Coerenta cache-ului in sisteme multiprocesor: protocolul MESI. TLB: functionarea, miss handling.	2	Prelegere interactiva	
Tema 5: Memoria virtuala. Paginarea: tabela de pagini, Page Table Entry (PTE), protectia. Tabele de pagini pe mai multe niveluri (x86-64: 4 niveluri). Mecanismul page fault. Segmentarea: istoricul, prezenta in x86. Huge pages: avantajele pentru aplicatii intensive. NUMA: accesul neuniform la memorie.	2	Prelegere interactiva	
Tema 6: Sistemul de intrare-iesire. Arhitectura I/O: controller, interfata, dispozitiv.	2	Prelegere interactiva	

Modalitatile de transfer: programat (polling), intreruperi, DMA. Ierarhia bus-urilor: PCIe (topologia, generatiile, latimea de banda). IOMMU: virtualizarea adreselor I/O. Interfete de stocare: NVMe, SATA, protocolul ATA.			
Tema 7: Aritmetica calculatoarelor. Reprezentarea intregilor: complement fata de 2, overflow. Reprezentarea in virgula mobila IEEE 754: single, double, NaN, Inf, denormalizate. Operatii pe virgula mobila: erorile de rotunjire, cancellation, standardul IEEE. Extensii SIMD: SSE2, AVX2, AVX-512: operatii vectoriale pe 128/256/512 biti.	2	Prelegere interactiva	
Tema 8: Setul de instructiuni x86-64. Registrele generale, segmentele, flagurile. Modurile de adresare: directa, indirecta, bazata cu index si scala. Conventia de apel (calling convention) System V AMD64 ABI: pasarea argumentelor, registrele salvate. Prologue si epilogue-ul unei functii. Instructiunile de control al fluxului.	2	Prelegere interactiva	
Tema 9: Optimizarea codului la nivel hardware. Efectele cache-ului asupra performantei: locality optimization, loop tiling, blocking. False sharing in sisteme multithread. Branch misprediction: masurarea cu perf, codul branch-free. Vectorizarea automata a compilatorului: conditiile, pragma SIMD. Profile-guided optimization (PGO).	2	Prelegere interactiva	
Tema 10: Procesoare GPU. Arhitectura GPU NVIDIA: SM, CUDA cores, warp, scheduler. Ierarhia memoriei GPU: registre, shared, L1/L2, global. Modelul de executie CUDA: grid, bloc, thread. Diferentele fata de CPU: throughput vs. latenta, paralelism masiv. Cazuri de utilizare: AI, HPC, grafică.	2	Prelegere interactiva	
Tema 11: Procesoare DSP. Cerintele specifice DSP: MAC in un ciclu, adresare circulara, bit-reversed. Arhitectura Harvard modificata. Unitatile MAC pe 32x32 biti cu acumulator extins. VLIW (Very Long Instruction Word): TMS320C6000. Comparatie DSP vs. FPGA vs. GPU pentru procesarea semnalelor.	2	Prelegere interactiva	
Tema 12: FPGA si procesoare configurabile. Arhitectura FPGA: LUT-uri, flip-flop-uri, DSP48, BRAM, interfete I/O. Fluxul de proiectare: HDL, sinteza, place-and-route, bitstream. HLS: C/C++ la RTL cu Vitis HLS. SoC FPGA (Zynq): ARM + FPGA pe acelasi chip. Comparatie ASIC vs. FPGA vs. GPU.	2	Prelegere interactiva	
Tema 13: Sisteme embedded si IoT. Microcontrolere vs. microprocesoare: diferentele, criteriile de alegere. ARM Cortex-M: profilele M0/M3/M4/M7, NVIC, SysTick. RTOS: FreeRTOS, Zephyr — task-uri, scheduling, semafoare. Interfete embedded:	2	Prelegere interactiva	

SPI, I2C, UART, CAN. Consumul de energie: modurile de sleep.			
Tema 14: Tendinte in arhitecturi de procesoare si recapitulare. Chiplet-uri: AMD EPYC, Intel Meteor Lake. Procesoare cu arhitectura heterogena: Apple M-series. Procesoare neuromorfice: Intel Loihi. RISC-V: ecosistemul, adoptia industrială. Calculul cuantic: qubiti, pori cuantice, limitele actuale. Recapitulare generala.	2	Prelegere, recapitulare	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> 1. Patterson, D.A., Hennessy, J.L., Computer Organization and Design: ARM Edition, 2nd ed., Morgan Kaufmann, 2020. 2. Hennessy, J.L., Patterson, D.A., Computer Architecture: A Quantitative Approach, 6th ed., Morgan Kaufmann, 2019. 3. Intel 64 and IA-32 Architectures Software Developer Manuals: https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Bryant, R.E., O'Hallaron, D.R., Computer Systems: A Programmer's Perspective, 3rd ed., Pearson, 2015. 5. ARM Architecture Reference Manual: https://developer.arm.com/documentation/ <p>Resurse software:</p> <ol style="list-style-type: none"> 6. Intel Software Developer Manuals: https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html 7. GEM5 Simulator Documentation: https://www.gem5.org/documentation/ 8. RISC-V ISA Specification: https://riscv.org/technical/specifications/ 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Pipeline si hazarduri in simulare. Simularea unui pipeline de 5 stadii in GEM5 sau intr-un simulator educational. Introducerea artificiala a hazardurilor RAW si observarea stallurilor generate. Activarea forwarding-ului si masurarea reducerii CPI.	2	Lucrare practica GEM5/simulator	
Lucrarea L2: Comportamentul cache-ului. Scrierea in C a doi algoritmi de parcurgere a unui tablou 2D: row-major vs. column-major. Masurarea timpilor de executie si a ratei de cache miss cu perf stat (Linux) sau VTune. Explicarea diferentei prin principiul localitatii.	2	Lucrare practica C, perf	
Lucrarea L3: Codul de asamblare x86-64. Compilarea unui program C simplu cu GCC la nivelele -O0, -O2 si -O3. Inspectarea codului de asamblare generat cu objdump sau Godbolt. Identificarea vectorizarii SIMD automate la -O3 si a eliminarii branch-urilor.	2	Lucrare practica GCC, objdump	
Lucrarea L4: Memoria virtuala si TLB. Scrierea unui program C care aloca pagini de memorie si masoara latentă accesului in functie de marimea working set-ului (curba latentă-marime). Identificarea pragurilor L1/L2/L3/RAM din grafic. Experimentarea cu huge pages (mmap + MAP_HUGETLB).	2	Lucrare practica C, Linux	
Lucrarea L5: Optimizarea codului — false sharing si branch prediction. Demonstrarea false sharing-ului: doua thread-uri care incrementeaza variabile adiacente in memorie vs. pe linii de cache separate. Masurarea degradarii performantei. Eliminarea unui	2	Lucrare practica C, pthreads	

branch cu operatii aritmetice si masurarea speedup-ului.			
Lucrarea L6: Profilarea cu Valgrind si perf. Profilarea unui program C cu Callgrind (Valgrind): identificarea functiei cu cel mai mare numar de instructiuni si de cache miss-uri. Generarea unui flame graph cu perf record si FlameGraph. Documentarea tehnica a bottleneck-urilor identificate.	2	Lucrare practica Valgrind, perf	
Lucrarea L7: Analiza comparativa a arhitecturilor. Realizarea unui raport tehnic comparativ (3-4 pagini) pentru un scenariu de procesare dat (ex. inferenta unui model CNN mic, procesarea unui filtru FIR pe 1 million esantioane, sau comprimarea unui fisier): selectarea si justificarea a trei arhitecturi candidate (CPU, GPU, DSP sau FPGA), estimarea performantei si a consumului de energie pentru fiecare, concluzia motivata privind arhitectura optima. Prezentarea raportului in fata grupei.	2	Raport tehnic, prezentare	
	<p>Bibliografie obligatorie laborator</p> <p>1. Patterson, D.A., Hennessy, J.L., Computer Organization and Design, ARM Edition, 2nd ed., Morgan Kaufmann, 2020.</p> <p>2. GEM5 Simulator Documentation: https://www.gem5.org/documentation/</p> <p>3. Valgrind Documentation: https://valgrind.org/docs/manual/</p> <p>Bibliografie complementara laborator</p> <p>4. Godbolt Compiler Explorer — inspectia codului de asamblare online: https://godbolt.org/</p> <p>5. FlameGraph — vizualizarea profilelor perf: https://www.brendangregg.com/flamegraphs.html</p> <p>Resurse online gratuite recomandate</p> <p>6. GEM5 Simulator: https://www.gem5.org/documentation/</p> <p>7. Valgrind Documentation: https://valgrind.org/docs/manual/</p> <p>8. Godbolt Compiler Explorer: https://godbolt.org/</p> <p>9. Intel VTune Profiler Documentation: https://www.intel.com/content/www/us/en/docs/vtune-profiler/</p>		

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

Calculatoare Numerice formeaza intelegerea profunda a hardware-ului pe care ruleaza orice software: de ce un algoritm scris corect poate fi de 10x mai lent din cauza cache miss-urilor, cum pipeline-ul genereaza hazarduri si cum compilatorul vectorizeaza codul. Aceasta perspectiva este esentiala pentru optimizarea aplicatiilor de performanta ridicata.

Raportul comparativ de arhitecturi din L7 formeaza direct A4 (documentatia tehnica conform standardelor ingineresti) si R4 (justificarea critica a alegerii tehnologiei in fata echipei), competente care diferentiaza un inginer capabil sa argumenteze deciziile hardware de unul care aplica solutii standard fara intelegerea implicatiilor.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoasterea organizarii pipeline-ului (stadiile, tipurile de hazarduri, forwarding);	Examen scris: analiza unui fragment de cod C si identificarea hazardurilor potentiale, calculul ratei de hit/miss pentru o	50%

	cunoasterea ierarhiei memoriei (cache set-associative, politici de inlocuire, coerenta MESI); cunoasterea reprezentarii IEEE 754 si a extensiilor SIMD; justificarea alegerii unei arhitecturi (CPU/GPU/DSP/FPGA) pentru un scenariu de procesare dat.	configuratie de cache data si justificarea alegerii arhitecturii optime pentru un scenariu descris	
8.5 Seminar/laborator	Corectitudinea masurarilor din laboratoare si interpretarea lor (explicarea diferentelor de performanta prin principii de arhitectura); calitatea raportului comparativ de arhitecturi din L7 (justificari cantitative, concluzie motivata).	Rapoarte de laborator predate pe Sakai dupa fiecare lucrare (50%) si raportul comparativ cu prezentare orala la L7 (50%)	50%
8.6 Standard minim de performanță:			
Raportul comparativ din L7 trebuie sa includa estimari cantitative de performanta (nu doar comparatii calitative) pentru cel puțin doua arhitecturi si o concluzie motivata pe baza datelor. Un raport care compara arhitecturi fara cifre concrete sau fara concluzie motivata nu indeplineste standardul minim.			

Data completării	Semnătura titularului de curs Conf. Dr. Bucea Manea Radu Țonis	Semnătura titularului de seminar / laborator Conf. Dr. Bucea Manea Radu Țonis
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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei				ELABORARE PROIECT DE DIPLOMĂ I– TIDS412			
1.2 Titularul activităților de curs							
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator / proiect / practică				Lect. Dr. Anghelache Iulia			
1.5 Anul de studiu	IV	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					25
Practică					
Elaborare proiect diplomă					25
2.8 Activități individuale					0
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	Toate disciplinele din An I–IV parcurse până la momentul elaborării; tema proiectului de diplomă stabilită în acord cu conducătorul științific.
3.2 de competențe	Capacitatea de a desfășura cercetare bibliografică independentă în baze de date academice (IEEE Xplore, ACM DL, Google Scholar); redactarea unui text tehnic structurat în LaTeX sau Word la standard academic; cunoașterea principiilor de bază ale managementului de proiect.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	
4.2 de desfășurare a seminarului/laboratorului	Activitatea se desfășoară sub îndrumarea conducătorului științific prin întâlniri periodice (față în față sau online); studentul lucrează individual la tema de diplomă; documentele intermediare și finale se predau pe platforma Sakai conform calendarului stabilit de facultate.

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 nivelul frontierelor cunoașterii actuale.
5.2 Aptitudini	A24. Elaborează studii de fezabilitate și analize tehnice comparative pentru evaluarea și selecția soluțiilor informatice optime, argumentând critic deciziile tehnice pe baza datelor cantitative și calitative disponibile. 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	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. R14. Inițiază și conduce activități de cercetare aplicată în domeniul TI, identificând în mod autonom direcții de investigație relevante, formulând ipoteze și validând experimental soluțiile propuse. R20. Gestionează autonom proiecte de inginerie software de complexitate ridicată, planificând activitățile, estimând și alocând resursele tehnice și umane necesare și adaptând planul de proiect la schimbările imprevizibile apărute pe parcurs. R21. Își asumă responsabilitatea pentru elaborarea studiilor de fezabilitate și a analizelor tehnice necesare fundamentării deciziilor strategice privind adoptarea de noi tehnologii sau arhitecturi informatice în cadrul organizației.

6. Conținuturi

6.1 Practică	Nr. ore	Metode de predare	Observații
Activitatea 1: Stabilirea temei și a planului de lucru. Discuția cu conducătorul științific pentru definirea precisă a temei, a obiectivelor și a ipotezei de lucru. Identificarea domeniului de cercetare, a problemei adresate și a contribuției originale planificate. Redactarea planului de lucru cu etape, livrabile și calendar.	5	Îndrumare individuală	
Activitatea 2: Cercetarea bibliografică și studiul literaturii. Căutarea sistematică a surselor relevante în IEEE Xplore, ACM Digital Library, Google Scholar și arXiv. Selectarea și lectura critică a minimum 20 de lucrări reprezentative. Redactarea unui rezumat adnotat (annotated bibliography) și identificarea lacunelor din literatura existentă.	5	Cercetare bibliografică	
Activitatea 3: Analiza comparativă a soluțiilor existente. Identificarea și compararea a minimum 3 soluții sau abordări existente pentru problema adresată. Analiza punctelor forte și a limitelor fiecărei soluții pe baza unor criterii tehnice definite. Justificarea direcției propuse față de alternativele analizate.	5	Analiză tehnică, îndrumare	

Activitatea 4: Proiectarea arhitecturii și a metodologiei. Proiectarea arhitecturii soluției propuse: diagrame de componente, diagrame de flux, modelul de date. Definierea metodologiei de cercetare sau de dezvoltare adoptate. Redactarea primelor capitole ale lucrării: introducere, stadiul artei, metodologie.	5	Proiectare, redactare	
Activitatea 5: Prezentarea intermediară și evaluarea progresului. Prezentarea stadiului lucrării în fața conducătorului și, după caz, a comisiei de monitorizare: tema, obiectivele, literatura studiată, arhitectura propusă, progresul față de calendarul planificat. Primirea feedback-ului și ajustarea planului pentru semestrul al doilea.	5	Prezentare intermediară	
<p>Bibliografie obligatorie practică</p> <ol style="list-style-type: none"> 1. Metodologia de elaborare a lucrărilor de licență — ghidul intern al Universității Internaționale Danubius, disponibil pe platforma Sakai. 2. IEEE Citation Reference Guide: https://ieeauthorcenter.ieee.org/ 3. Google Scholar: https://scholar.google.com/ <p>Bibliografie complementara practică</p> <ol style="list-style-type: none"> 4. IEEE Xplore Digital Library: https://ieeexplore.ieee.org/ 5. ACM Digital Library: https://dl.acm.org/ <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. Google Scholar: https://scholar.google.com/ 7. IEEE Xplore: https://ieeexplore.ieee.org/ 8. ACM Digital Library: https://dl.acm.org/ 9. arXiv — preprint-uri în domenii tehnice: https://arxiv.org/ 			

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

Elaborarea Proiectului de Diplomă I marchează tranziția studentului de la receptor de cunoaștere la producător de cunoaștere tehnică. Cele 25 de ore dirijate formează fundamentul lucrării finale: tema definită riguros, literatura sistematic studiată, arhitectura proiectată și justificată critic. Activitățile formează direct C25 (cunoaștere integrată și critică a domeniului), A24 (analiza comparativă a soluțiilor), R13 (documentarea riguroasă) și R14 (cercetarea aplicată autonomă), competențe care nu pot fi formate în cadrul disciplinelor de curs și laborator.

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	Calitatea planului de lucru (obiective SMART, calendar realist); calitatea rezumatului adnotat (minimum 20 de surse relevante, analiza critică a fiecăreia); calitatea analizei comparative a soluțiilor existente (criterii tehnice definite, justificarea direcției	Evaluare de către conducătorul științific: plan de lucru (20%) + rezumat adnotat (30%) + analiză comparativă (30%) + prezentare intermediară (20%)	100%

	propuse); calitatea prezentării intermediare de la Activitatea 5.		
--	---	--	--

8.6 Standard minim de performanță:

Predarea unui rezumat adnotat cu minimum 20 de surse bibliografice relevante și a unui document de analiză comparativă care justifică direcția propusă față de cel puțin 3 soluții existente. Lipsa analizei comparative sau un rezumat adnotat cu mai puțin de 10 surse nu îndeplinesc standardul minim și blochează continuarea în semestrul al doilea.

Data completării	Semnătura titularului de curs	Semnătura titularului de seminar / laborator/proiect/practică Lect. Dr. Anghelache Iulia
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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei		SISTEME DE PROGRAMARE PENTRU TIMP REAL – TIDS414					
1.2 Titularul activităților de curs		Conf. Dr. Popescu Florin					
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator		Conf. Dr. Popescu Florin					
1.5 Anul de studiu	IV	1.6 Semestrul	1	1.7 Tipul de evaluare	E	1.8 Regimul disciplinei	OPT

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					20
Referate					16
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	Sisteme de Operare (TIDS302) — procese, fire de execuție, sincronizare; Algoritmi Paraleli și Distribuți (TIDS315); Arhitectura Sistemelor de Calcul (TIDS214).
3.2 de competențe	Cunoașterea mecanismelor de sincronizare (mutex, semafor, monitor); programare C sau C++ la nivel mediu; înțelegerea conceptului de latență deterministă și a diferenței față de sistemele time-sharing.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu videoproiector și acces la internet; platforma Sakai; recomandare: FreeRTOS sau Zephyr instalat pe o placă de dezvoltare (STM32 Nucleo sau Raspberry Pi Pico) sau în emulator QEMU.
4.2 de desfășurare a seminarului/laboratorului	Laborator cu calculatoare individuale; Linux cu PREEMPT_RT patch sau RTOS în QEMU; plăci STM32 Nucleo sau Raspberry Pi Pico cu FreeRTOS; osciloscop digital pentru măsurarea jitter-ului; studenții pot folosi propriul laptop cu WSL2 sau Linux nativ.

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

5.1 Cunoștințe	C23. Cunoaște în profunzime protocoalele avansate de comunicații și arhitecturile sistemelor de programare pentru timp real, cu conștientizarea critică a cerințelor de performanță și fiabilitate în sisteme critice.
5.2 Aptitudini	A8. Implementează algoritmi paraleli și distribuiți, valorificând arhitecturile moderne de procesoare multi-core și infrastructurile de calcul distribuit pentru optimizarea performanței sistemelor.
5.3 Responsabilitate și autonomie	R23. Își asumă responsabilitatea pentru înțelegerea corectă și completă a cerințelor beneficiarilor, interacționând proactiv cu utilizatorii și factorii de decizie pentru clarificarea ambiguităților și validarea soluțiilor propuse pe parcursul întregului proces de dezvoltare.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Sisteme de timp real — definiții și clasificare. Sistemele hard real-time vs. soft real-time vs. firm real-time. Parametrii temporali: perioadă, termen limită (deadline), timp de execuție (WCET). Exemple industriale: controlere PLC, sisteme avionice DO-178C, sisteme medicale IEC 62304.	2	Prelegere interactivă	
Tema 2: Modelul de execuție și WCET. Analiza timpului de execuție în cel mai rău caz (WCET): metode statice și de măsurare. Factori care influențează WCET: cache, pipeline, branch prediction. Instrumente: aiT, OTAWA. Programarea fără recursivitate și fără alocare dinamică.	2	Prelegere interactivă	
Tema 3: Planificarea sarcinilor — algoritmi preemptivi. Modelul taskului periodic: (C, T, D). Algoritmul Rate Monotonic (RM): condiția de schedulabilitate (Liu & Layland). Algoritmul Earliest Deadline First (EDF): optimalitatea pentru sisteme preemptive. Compararea RM vs. EDF.	2	Prelegere interactivă	
Tema 4: Planificarea sarcinilor — algoritmi nepreemptivi și aperiodici. Planificarea nepreemptivă: avantajele pentru sisteme simple. Taskuri aperiodice: algoritmul Background Scheduling, Polling Server, Sporadic Server. Taskuri sporadice: modelul, garantarea deadlineurilor.	2	Prelegere interactivă	
Tema 5: Protocoale de acces la resurse partajate. Problema inversiunii de prioritate. Priority Inheritance Protocol (PIP): mecanismul, limitele. Priority Ceiling Protocol (PCP): ceiling-ul resursei, blocarea garantată. Stack Resource Policy (SRP). Compararea protocoalelor în sisteme embedded.	2	Prelegere interactivă	
Tema 6: Sisteme de operare pentru timp real — RTOS. Structura unui RTOS: kernel, scheduler, driver-e. FreeRTOS: task-uri (xTaskCreate), priorități, preemptare. Zephyr	2	Prelegere interactivă	

RTOS: arhitectura, configurarea (Kconfig), device tree. VxWorks și QNX: utilizarea în sisteme critice. Comparatie RTOS vs. Linux cu PREEMPT RT.			
Tema 7: Mecanisme de sincronizare în RTOS. Mutexuri în FreeRTOS: xSemaphoreCreateMutex, priority inheritance. Semafoare de numărare: producător-consumator în RTOS. Queue-uri: comunicarea între taskuri, notificările directe (task notifications). Timer-e software: xTimerCreate, one-shot vs. auto-reload.	2	Prelegere interactivă	
Tema 8: Întreruperi și gestionarea lor. Latența de întrerupere: definiție, măsurare. ISR (Interrupt Service Routine): regulile (timp scurt, fără blocare). Mecanismul deferred interrupt handling: taskuri ISR-deferred în FreeRTOS. Nivelele de prioritate hardware vs. prioritățile RTOS. Gestionarea în Linux PREEMPT RT: threaded interrupts.	2	Prelegere interactivă	
Tema 9: Comunicații în sisteme de timp real. CAN bus: arbitrarea pe prioritate, latența deterministă, utilizarea în automotive (ISO 11898). TT-CAN și FlexRay: protocoalele time-triggered. EtherCAT: sincronizarea distribuită, latența sub milisecundă. Compararea cu protocoalele IT (TCP/IP) din perspectiva determinismului.	2	Prelegere interactivă	
Tema 10: Linux cu PREEMPT_RT. Kernel Linux standard vs. PREEMPT_RT patch: diferențele de latență. Configurarea pentru timp real: izolarea CPU-urilor (isolcpus), dezactivarea tickless, afinitatea firelor. Măsurarea latentei cu cyclictst. Compararea cu un RTOS dedicat pe același hardware.	2	Prelegere interactivă	
Tema 11: Toleranța la defecte în sisteme critice. Clasificarea defectelor: transiente, permanente, intermitente. Redundanța hardware: Triple Modular Redundancy (TMR). Watchdog timer: tipurile, configurarea, resetul la blocaj. Mecanismele de detecție a erorilor: CRC, ECC pentru memorie. Standardele IEC 61508 și DO-178C.	2	Prelegere interactivă	
Tema 12: Sisteme de timp real distribuite. Sincronizarea ceasurilor în sisteme distribuite: PTP (IEEE 1588), gPTP (IEEE 802.1AS). Time-Sensitive Networking (TSN): standardele 802.1Qbv (time-aware shaping), 802.1Qbu (frame preemption). Aplicații: Ethernet industrial, ADAS automotive.	2	Prelegere interactivă	
Tema 13: Verificarea și validarea sistemelor de timp real. Model checking: UPPAAL — rețele de automate temporale, specificarea și verificarea proprietăților de timp real. Simularea RTOS cu QEMU: validarea schedulingului. Testarea temporală: injectarea	2	Prelegere interactivă	

de perturbații, worst-case testing. Certificarea DO-178C DAL A.			
Tema 14: Tendințe și recapitulare. Sisteme de timp real pe procesoare multi-core: partajarea resurselor, interferența pe bus. Mixed-criticality systems (MCS): taskuri cu niveluri diferite de criticalitate pe același procesor. AUTOSAR: arhitectura, rolul în automotive. Recapitulare generală.	2	Prelegere, recapitulare	
	<p>Bibliografie obligatorie</p> <p>1. Liu, J.W.S., Real-Time Systems, Prentice Hall, 2000.</p> <p>2. Burns, A., Wellings, A., Real-Time Systems and Programming Languages, 4th ed., Addison-Wesley, 2009.</p> <p>3. FreeRTOS Documentation: https://www.freertos.org/Documentation/RTOS_book.html</p> <p>Bibliografie complementară</p> <p>4. Buttazzo, G., Hard Real-Time Computing Systems, 3rd ed., Springer, 2011.</p> <p>5. Zephyr RTOS Documentation: https://docs.zephyrproject.org/</p> <p>Resurse software:</p> <p>6. FreeRTOS Documentation: https://www.freertos.org/Documentation/RTOS_book.html</p> <p>7. Zephyr RTOS Documentation: https://docs.zephyrproject.org/</p> <p>8. UPPAAL Model Checker: https://uppaal.org/</p>		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Primul task FreeRTOS. Crearea a două taskuri FreeRTOS cu priorități diferite în QEMU sau pe STM32. Observarea preemptării: taskul cu prioritate mai mare întrerupe taskul inferior. Măsurarea perioadei de execuție cu vTaskDelayUntil.	2	Lucrare practică FreeRTOS	
Lucrarea L2: Sincronizare cu mutex și priority inheritance. Reproducerea inversiunii de prioritate cu trei taskuri și o resursă partajată. Activarea priority inheritance și observarea eliminării inversiunii. Măsurarea timpului de blocare cu și fără PIP.	2	Lucrare practică FreeRTOS	
Lucrarea L3: Comunicare prin queue-uri. Implementarea unui sistem producător-consumator cu queue FreeRTOS: un task generează date senzori, altul le procesează. Testarea comportamentului la overflow al queue-ului (bloc vs. drop). Utilizarea task notifications ca alternativă lightweight.	2	Lucrare practică FreeRTOS	
Lucrarea L4: Schedulabilitate Rate Monotonic. Calculul utilizării CPU pentru trei seturi de taskuri periodice cu formula Liu & Layland. Verificarea schedulabilității. Simularea în QEMU și confirmarea că deadline-urile sunt respectate sau nu, conform analizei teoretice.	2	Lucrare practică Python, QEMU	
Lucrarea L5: Măsurarea jitter-ului și a latenței. Măsurarea latenței de întrerupere pe Linux standard vs. Linux PREEMPT_RT cu cyclicttest. Configurarea izolării CPU (isolcpus). Analiza histogramei jitter-ului: compararea celor două configurații și identificarea cauzelor vârfurilor.	2	Lucrare practică Linux, cyclicttest	

Lucrarea L6: Watchdog și toleranță la defecte. Configurarea unui hardware watchdog pe STM32 sau în QEMU. Simularea blocării unui task (buclă infinită) și observarea resetului sistemului. Implementarea unui pattern de refreshal watchdog distribuit între taskuri cu token. Testarea recuperării după reset.	2	Lucrare practică FreeRTOS, STM32	
Lucrarea L7: Studiu de caz — analiza unui sistem de timp real existent. Analiza unui sistem open-source de timp real real (autopilot ArduPilot sau PX4, controler CNC grbl, sau sistemul de control al unui braț robotic cu ROS2). Identificarea taskurilor, priorităților, deadline-urilor și protocoalelor de sincronizare utilizate. Redactarea unui raport tehnic cu diagrama de scheduling și evaluarea critică a alegerilor de proiectare.	2	Analiză, raport tehnic	
<p>Bibliografie obligatorie laborator</p> <p>1. FreeRTOS Documentation: https://www.freertos.org/Documentation/RTOS_book.html</p> <p>2. Zephyr RTOS Documentation: https://docs.zephyrproject.org/</p> <p>3. Burns, A., Wellings, A., Real-Time Systems and Programming Languages, 4th ed., Addison-Wesley, 2009.</p> <p>Bibliografie complementara laborator</p> <p>4. UPPAAL Documentation: https://uppaal.org/</p> <p>5. cyclicttest Documentation — măsurarea latenței Linux RT: https://wiki.linuxfoundation.org/realtime/documentation/howto/tools/cylicttest/start</p> <p>Resurse online gratuite recomandate</p> <p>6. FreeRTOS Documentation: https://www.freertos.org/Documentation/RTOS_book.html</p> <p>7. Zephyr RTOS Documentation: https://docs.zephyrproject.org/</p> <p>8. Linux PREEMPT_RT Wiki: https://wiki.linuxfoundation.org/realtime/start</p> <p>9. cyclicttest Documentation: https://wiki.linuxfoundation.org/realtime/documentation/howto/tools/cylicttest/start</p>			

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

Sisteme de Programare pentru Timp Real acoperă competențele esențiale pentru industria embedded și automotive: planificarea sarcinilor cu garanții temporale, RTOS, protocoale deterministe și toleranța la defecte. Aceste competențe sunt solicitate explicit în sectoarele automotive (AUTOSAR), avionic (DO-178C) și industrial (IEC 61508).

Studiul de caz din L7 — analiza critică a alegerilor de proiectare dintr-un sistem real — formează direct R23: asumarea responsabilității pentru înțelegerea completă a cerințelor și validarea soluțiilor propuse față de beneficiari.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea modelului taskului periodic și a algoritmilor RM și EDF cu analiza schedulabilității; cunoașterea mecanismelor de sincronizare (PIP, PCP)	Examen scris: calculul schedulabilității RM pentru un set de taskuri dat, identificarea inversiunii de prioritate într-un scenariu descris și alegerea protocolului de comunicație cu justificare	40%

	și a inversiunii de prioritate; identificarea protocolului de comunicație adecvat (CAN, EtherCAT, TSN) pentru un scenariu dat; cunoașterea mecanismelor de toleranță la defecte (watchdog, TMR, ECC).		
8.5 Seminar/laborator	Corectitudinea implementărilor FreeRTOS (taskurile respectă deadline-urile, sincronizarea elimină inversiunea de prioritate); calitatea raportului tehnic din L7 (diagrama de scheduling corectă, evaluarea critică a alegerilor de proiectare).	Rapoarte de laborator predate pe Sakai după fiecare lucrare (50%) și raportul tehnic din L7 cu prezentare orală (50%)	60%
8.6 Standard minim de performanță:			
Demonstrarea la lucrarea L2 că inversiunea de prioritate este reproductibilă fără PIP și că dispăre cu PIP activat, cu măsurători concrete ale timpului de blocare. Un student care nu poate reproduce și explica inversiunea de prioritate nu a înțeles nucleul disciplinei și nu îndeplinește standardul minim.			

Data completării	Semnătura titularului de curs	Semnătura titularului de seminar / laborator
16.03.2026	Conf. Dr. Popescu Florin	Conf. Dr. Popescu Florin
Data avizării în Consiliul Departamentului	Semnătura directorului DIT	
24.03.2026	Lect. dr. Antohe Valerian	
Data aprobării în Consiliul Facultății	Semnătura decanului FMDT	
07.04.2026	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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei	PARADIGME DE PROGRAMARE – TIDS415						
1.2 Titularul activităților de curs	Conf. Dr. Popescu Florin						
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator	Conf. Dr. Popescu Florin						
1.5 Anul de studiu	IV	1.6 Semestrul	1	1.7 Tipul de evaluare	E	1.8 Regimul disciplinei	OPT

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					20
Referate					16
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	Proiectarea Algoritmilor (TIDS307); Ingineria Programelor (TIDS308); Limbaje Formale și Translatoare (TIDS310).
3.2 de competențe	Cunoașterea programării orientate pe obiecte (clase, moștenire, polimorfism); programare Python sau Haskell la nivel introductiv; înțelegerea noțiunilor de funcție pură și efect lateral.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu videoprojector și acces la internet; platforma Sakai; recomandare: GHCup (compilerul Haskell) și Racket instalate înainte de curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator cu calculatoare individuale; GHC și GHCi (Haskell), Racket, Python 3.10+; VS Code cu extensiile Haskell Language Server și Racket; 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	C23. Cunoaște în profunzime protocoalele avansate de comunicații și arhitecturile sistemelor de programare pentru timp real, cu conștientizarea critică a cerințelor de performanță și fiabilitate în sisteme critice.
5.2 Aptitudini	A8. Implementează algoritmi paraleli și distribuți, valorificând arhitecturile moderne de procesoare multi-core și infrastructurile de calcul distribuit pentru optimizarea performanței sistemelor.
5.3 Responsabilitate și autonomie	R23. Își asumă responsabilitatea pentru înțelegerea corectă și completă a cerințelor beneficiarilor, interacționând proactiv cu utilizatorii și factorii de decizie pentru clarificarea ambiguităților și validarea soluțiilor propuse pe parcursul întregului proces de dezvoltare.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Panorama paradigmelor de programare. Clasificarea paradigmelor: imperativă, orientată pe obiecte, funcțională, logică, concurentă, reactivă. Criteriile de comparație: expresivitatea, siguranța tipurilor, modelul de execuție, potrivirea cu domeniul problemei. Limbaje multi-paradigmă: Scala, Rust, Python.	2	Prelegere interactivă	
Tema 2: Programarea funcțională — fundamente. Funcții pure și efecte laterale. Imutabilitatea datelor. Funcțiile de ordin superior: map, filter, fold/reduce. Transparența referențială. Evaluarea leneșă (lazy evaluation) vs. strictă. Comparație cu programarea imperativă pe exemple concrete.	2	Prelegere interactivă	
Tema 3: Lambda calcul și tipuri. Lambda calcul: sintaxa, reducerile alfa și beta, forma normală. Combinator fix (recursie fără nume). Sistemul de tipuri Hindley-Milner: inferența automată de tipuri. Tipuri polimorfice: parametric vs. ad-hoc. Clasele de tipuri Haskell: Eq, Ord, Show, Num.	2	Prelegere interactivă	
Tema 4: Haskell — tipuri algebrice și pattern matching. Tipuri de date algebrice (ADT): Sum types (Either, Maybe), Product types (tuple, record). Pattern matching exhaustiv: cazurile acoperite, wildcard. Tipul Maybe pentru gestionarea valorilor absente. Funcțiile recursive pe liste cu pattern matching.	2	Prelegere interactivă	
Tema 5: Haskell — functorii, aplicativii și monadele. Functor: fmap, legile. Applicative: pure, (<*>), independența efectelor. Monad: (>>=), (>>), legile monadice. Monada Maybe: înlănțuirea calculelor cu eșec. Monada IO: separarea efectelor laterale de codul pur. do-notation.	2	Prelegere interactivă	
Tema 6: Haskell — monade utile și transformatoare. Monada State: calculul cu stare imutabilă simulată. Monada Reader: configurația partajată fără parametri globali.	2	Prelegere interactivă	

Monada Writer: acumularea logurilor. Transformatoarele de monade: MaybeT, StateT — stivuirea efectelor. Monad-ul Except pentru gestionarea erorilor.			
Tema 7: Programarea logică — Prolog. Modelul de execuție Prolog: fapte, reguli, interogări. Unificarea și backtracking-ul. Structuri de date în Prolog: liste, arbori. Tăietura (cut): controlul backtracking-ului. Negarea prin eșec. Aplicații: sisteme expert simple, verificarea constrângerilor.	2	Prelegere interactivă	
Tema 8: Programarea bazată pe constrângeri. Programarea prin constrângeri (CLP): variabile, domenii, constrângeri. CLP(FD) în SWI-Prolog: label, indomain, propagarea constrângerilor. Diferența față de programarea logică clasică. Aplicații: sudoku, planificarea orarelor, problema n-reginilor.	2	Prelegere interactivă	
Tema 9: Programarea reactivă. Programarea reactivă: fluxuri de date (streams), propagarea schimbărilor. Observable-uri: crearea, transformarea (map, filter, flatMap), combinarea (merge, zip). RxJS și RxPY: exemple practice. Backpressure: gestionarea fluxurilor rapide. Diferența față de programarea eveniment-driven clasică.	2	Prelegere interactivă	
Tema 10: Programarea concurentă funcțională. Modelul actorilor: actori, mesaje, cutii poștale. Erlang/Elixir: procesele lightweight, trimiterea mesajelor, supravegherea proceselor (supervisors). Software Transactional Memory (STM) în Haskell: TVar, atomically, retry. Comparatie modele de concurență: actori vs. STM vs. canale (Go).	2	Prelegere interactivă	
Tema 11: Metaprogramare și macros. Metaprogramarea: codul care generează cod. Macros Lisp/Racket: quasiquote, syntax-rules, defmacro — manipularea arborelui sintactic. Template metaprogramming C++: exemple. Metaprogramare în Elixir: quote/unquote. Limitele și riscurile metaprogramării: lisibilitate, depanare.	2	Prelegere interactivă	
Tema 12: Programarea orientată pe tipuri. Type-Driven Development: tipurile ca specificații. Tipuri dependente: Idris — funcțiile la nivel de tipuri, vectorii cu lungime garantată în tipuri. Tipuri liniare: Rust — ownership, borrow checker, siguranța memoriei fără garbage collector. Phantom types: prevenirea erorilor la compilare.	2	Prelegere interactivă	
Tema 13: Programarea funcțională în limbaje mainstream. Funcționalele în Python: functools, itertools, comprehensions, generators. Programarea funcțională în Java: lambdas, Stream API, Optional, metode de	2	Prelegere interactivă	

referință. Programarea funcțională în Scala: case classes, pattern matching, for-comprehensions, implicits. Avantajele stilului funcțional în cod de producție.			
Tema 14: Alegerea paradigmei și recapitulare. Criteriile de alegere a paradigmei: natura problemei, siguranța, performanța, ecosistemul. Domeniile de excelență: funcțional pentru transformări de date și concurență, logic pentru raționament și constrângeri, reactiv pentru UI și streaming. Tendințe: limbajele funcționale în industrie (Haskell la Facebook, Erlang la WhatsApp, Elixir la Discord). Recapitulare generală.	2	Prelegere, recapitulare	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> 1. Bird, R., Thinking Functionally with Haskell, Cambridge University Press, 2014. 2. Lipovaca, M., Learn You a Haskell for Great Good!, No Starch Press, 2011. Disponibil gratuit la: https://learnyouahaskell.com/ 3. Haskell Documentation: https://www.haskell.org/documentation/ <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Sterling, L., Shapiro, E., The Art of Prolog, 2nd ed., MIT Press, 1994. 5. SWI-Prolog Documentation: https://www.swi-prolog.org/pldoc/ <p>Resurse software:</p> <ol style="list-style-type: none"> 6. Haskell Documentation: https://www.haskell.org/documentation/ 7. Learn You a Haskell — carte gratuită: https://learnyouahaskell.com/ 8. SWI-Prolog Documentation: https://www.swi-prolog.org/pldoc/ 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Haskell — tipuri și funcții de bază. Definierea tipurilor algebrice proprii (ADT) în GHCi: un tip pentru expresii aritmetice (Lit, Add, Mul). Implementarea unui evaluator prin pattern matching recursiv. Utilizarea Maybe pentru gestionarea împărțirii la zero.	2	Lucrare practică Haskell/GHCi	
Lucrarea L2: Haskell — funcții de ordin superior și liste. Implementarea propriilor versiuni de map, filter și foldr. Rezolvarea a 5 probleme pe liste exclusiv cu funcții de ordin superior (fără recursie explicită). Compararea lizibilității cu varianta imperativă echivalentă în Python.	2	Lucrare practică Haskell	
Lucrarea L3: Haskell — monada Maybe și IO. Implementarea unui parser simplu de expresii aritmetice cu monada Maybe: eșecul la întâlnirea unui simbol neașteptat. Rescrierea cu do-notation. Utilizarea monadei IO pentru citirea și scrierea fișierelor.	2	Lucrare practică Haskell	
Lucrarea L4: Prolog — fapte, reguli și interogări. Implementarea unei baze de cunoștințe despre relații familiale (tată, mamă, bunic, unchi). Interogări recursive: găsirea tuturor descendenților unui nod. Implementarea algoritmilor de sortare (quicksort, mergesort) în Prolog.	2	Lucrare practică SWI-Prolog	
Lucrarea L5: Prolog — constrângeri CLP(FD). Rezolvarea problemei n-reginelor și a sudoku-ului cu CLP(FD) în SWI-Prolog. Compararea timpului de rezolvare cu	2	Lucrare practică SWI-Prolog	

backtracking pur vs. propagarea constrângerilor. Rezolvarea unui problem de planificare simplă.			
Lucrarea L6: Programare reactivă cu RxPY. Implementarea unui pipeline de procesare a datelor cu RxPY: un flux de evenimente simulate (senzori), filtrarea, transformarea și agregarea cu fereastră glisantă (window). Gestionarea erorilor în flux cu catch.	2	Lucrare practică Python, RxPY	
Lucrarea L7: Comparatie inter-paradigmă pe aceeași problemă. Implementarea aceleiași probleme (ex. interpretorul unui mini-limbaj de expresii sau un sistem de recomandare bazat pe reguli) în trei paradigme: imperativ (Python), funcțional (Haskell) și logic (Prolog). Redactarea unui raport tehnic comparativ: volumul de cod, lizibilitatea, ușurința de extindere, performanța.	2	Comparatie, raport tehnic	
	<p>Bibliografie obligatorie laborator</p> <p>1. Haskell Documentation: https://www.haskell.org/documentation/</p> <p>2. SWI-Prolog Documentation: https://www.swi-prolog.org/pldoc/</p> <p>3. Lipovaca, M., Learn You a Haskell for Great Good! Disponibil gratuit la: https://learnyouahaskell.com/</p> <p>Bibliografie complementara laborator</p> <p>4. RxPY Documentation: https://rxpy.readthedocs.io/</p> <p>5. Racket Documentation: https://docs.racket-lang.org/</p> <p>Resurse online gratuite recomandate</p> <p>6. Learn You a Haskell: https://learnyouahaskell.com/</p> <p>7. SWI-Prolog Documentation: https://www.swi-prolog.org/pldoc/</p> <p>8. RxPY Documentation: https://rxpy.readthedocs.io/</p> <p>9. Hoogle — motorul de căutare Haskell după tipuri: https://hoogle.haskell.org/</p>		

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

<p>Paradigmele de Programare formează capacitatea de a alege instrumentul conceptual potrivit problemei, nu de a aplica mecanic o singură paradigmă. Studiul Haskell, Prolog și RxPY în același semestru expune studentul la trei moduri radical diferite de a gândi o soluție, lărgind vocabularul ingineresc dincolo de OOP și programarea imperativă.</p> <p>Comparația inter-paradigmă din L7 formează direct R23: responsabilitatea înțelegerii complete a cerințelor problemei înainte de a alege paradigma, și capacitatea de a justifica alegerea față de beneficiari cu argumente concrete.</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 paradigmelor funcțională, logică și reactivă; implementarea corectă a monadelor Maybe și IO în Haskell; rezolvarea unei probleme de constrângeri în Prolog CLP(FD); justificarea	Examen scris: implementarea unei funcții Haskell cu tipul dat (inferența tipului), rezolvarea unei interogări Prolog pe o bază de cunoștințe furnizată și argumentarea alegerii paradigmei potrivite pentru o problemă descrisă	60%

	alegerii paradigmei pentru un scenariu dat.		
8.5 Seminar/laborator	Corectitudinea implementărilor Haskell și Prolog din laborator; calitatea raportului comparativ din L7 (cele trei implementări funcționale, analiza obiectivă a avantajelor și limitelor fiecărei paradigme).	Implementări predate pe Sakai după fiecare lucrare (50%) și raport comparativ cu prezentare orală la L7 (50%)	40%
8.6 Standard minim de performanță:			
Raportul din L7 trebuie să conțină trei implementări funcționale ale aceleiași probleme (imperativ, funcțional, logic) cu o analiză comparativă bazată pe criterii concrete (linii de cod, lizibilitate, extensibilitate). Un raport cu o singură implementare sau fără comparație obiectivă nu îndeplinește standardul minim.			

Data completării 17.03.2026	Semnătura titularului de curs Conf. Dr. Popescu Florin	Semnătura titularului de seminar / laborator Conf. Dr. Popescu Florin
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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei		MACHINE LEARNING – TIDS416					
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	IV	1.6 Semestrul	1	1.7 Tipul de evaluare	E	1.8 Regimul disciplinei	OPT

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					7
Referate					5
Teme casă					
Proiect					
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	Inteligență Artificială (TIDS403); Matematici Aplicate (TIDS206) — algebră liniară, calcul diferențial, probabilități; Baze de Date (TIDS305).
3.2 de competențe	Cunoașterea algoritmilor ML de bază (regresie, clasificare, clustering) din IA; programare Python cu scikit-learn și PyTorch la nivel mediu; înțelegerea noțiunilor de gradient, funcție de cost și backpropagation.

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 Google Colab sau Kaggle Notebooks activ înainte de curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator cu calculatoare individuale sau acces la Google Colab cu GPU; Python cu PyTorch, scikit-learn, Hugging Face Transformers, MLflow, SHAP; 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	C17. Cunoaște în profunzime principiile inteligenței artificiale — algoritmi de căutare, reprezentarea cunoștințelor, sisteme expert — și metodele de învățare automată (Machine Learning), cu conștientizarea critică a limitelor și implicațiilor etice ale acestora. C19. Posedă cunoștințe avansate de inteligență computațională integrată — rețele neuronale, algoritmi evolutivi, logică fuzzy — cu capacitatea de a evalua critic aplicabilitatea acestora în rezolvarea problemelor ingineresti complexe.
5.2 Aptitudini	A17. Proiectează și implementează sisteme de inteligență artificială și modele de învățare automată (Machine Learning), evaluând critic performanța acestora și adaptând hiperparametrii pentru optimizarea rezultatelor.
5.3 Responsabilitate și autonomie	R17. Acționează în mod responsabil și etic în toate activitățile profesionale din domeniul TI, respectând drepturile de proprietate intelectuală, legislația privind protecția datelor cu caracter personal (GDPR) și standardele de conduită profesională.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Fundamente avansate ale ML. Recapitulare: bias-variance tradeoff, regularizare L1/L2, validare încrucișată. Distribuțiile de probabilitate în ML: gaussiană, multinomială, Bernoulli. Estimarea maximă a verosimilității (MLE) și estimatorul MAP. Teorema Bayes în clasificare.	2	Prelegere interactivă	
Tema 2: Metode kernel și SVM avansat. Kernelul RBF: interpretarea geometrică, parametrul gamma. SVM multi-clasă: one-vs-rest, one-vs-one. Kernelurile personalizate: string kernel, graph kernel. Kernel PCA: reducerea dimensionalității neliniare.	2	Prelegere interactivă	
Tema 3: Metode de ansamblu avansate. Gradient Boosting: algoritmul general, funcția de pierdere, regularizarea. XGBoost: aproximarea de ordinul 2, regularizarea L1/L2, gestionarea valorilor lipsă. LightGBM: leaf-wise growth, feature binning. CatBoost: codarea categorică ordonată. Compararea pe benchmark-uri Kaggle.	2	Prelegere interactivă	
Tema 4: Rețele neuronale convoluționale avansate. Arhitecturi moderne: EfficientNet (compound scaling), Vision Transformer (ViT): patch embeddings, self-attention pentru imagini. CLIP: alinierea text-imagine. Augmentarea avansată: MixUp, CutMix, AutoAugment. Knowledge distillation.	2	Prelegere interactivă	
Tema 5: Rețele recurente și modele de secvențe. LSTM avansat: peephole connections, bidirectional LSTM. GRU. Seq2Seq cu attention: mecanismul Bahdanau, aplicații traducere. Wav2Vec 2.0: preantrenarea auto-supervizată pentru audio. Time series forecasting: N-BEATS, Temporal Fusion Transformer.	2	Prelegere interactivă	

Tema 6: Transformare și modele de limbaj mari. Arhitectura Transformer completă: multi-head attention, positional encoding, feed-forward. BERT: preantrenarea (MLM, NSP), fine-tuning. GPT: modelul autoregresiv, few-shot learning. T5: text-to-text framework. Eficiența: DistilBERT, ALBERT, Longformer.	2	Prelegere interactivă	
Tema 7: Modele generative. Variational Autoencoder (VAE): ELBO, reparametrizarea, spațiul latent. GAN: discriminatorul, generatorul, funcția minimax, instabilitatea antrenării. Îmbunătățiri GAN: DCGAN, StyleGAN, Wasserstein GAN. Modele de difuzie (DDPM): procesul forward și reverse, antrenarea.	2	Prelegere interactivă	
Tema 8: Reinforcement Learning avansat. Policy gradient: REINFORCE, baseline. Actor-Critic: A3C, A2C. Proximal Policy Optimization (PPO): clipping-ul, antrenarea stabilă. AlphaGo/AlphaZero: MCTS cu rețele neuronale. RL din feedback uman (RLHF): aplicarea în LLM-uri.	2	Prelegere interactivă	
Tema 9: Învățare automată pe grafuri. Grafuri ca date: noduri, muchii, atribute. Graph Neural Networks (GNN): message passing. Graph Convolutional Networks (GCN). GraphSAGE: eșantionarea vecinilor. Graph Attention Networks (GAT). Aplicații: rețele sociale, chimie moleculară, sisteme de recomandare.	2	Prelegere interactivă	
Tema 10: Explicabilitatea modelelor ML. Necesitatea XAI: reglementările EU AI Act. Metode locale: LIME (aproximarea liniară locală), SHAP (valori Shapley). Metode globale: importanța caracteristicilor prin permutare, partial dependence plots. Grad-CAM pentru CNN. Limitele explicabilității: post-hoc vs. intrinsec.	2	Prelegere interactivă	
Tema 11: MLOps și ciclul de viață al modelelor. Versioning date și modele: DVC, MLflow Tracking. Model registry: MLflow Models. Serving: FastAPI + Docker, TorchServe, Triton Inference Server. Monitorizarea modelelor în producție: data drift (Evidently), concept drift. CI/CD pentru ML: GitHub Actions cu antrenare automată.	2	Prelegere interactivă	
Tema 12: Scalabilitatea antrenării ML. Antrenarea distribuită: data parallelism (DDP PyTorch), model parallelism (pipeline, tensor). Mixed precision training: FP16, BF16, scaler-ul gradientului. Gradient checkpointing. Quantization: INT8, INT4. LoRA (Low-Rank Adaptation): fine-tuning eficient al LLM-urilor.	2	Prelegere interactivă	

Tema 13: ML aplicat în domenii specifice. Computer vision aplicată: object detection (YOLOv8), segmentare semantică (SAM), depth estimation. Procesarea limbajului natural aplicată: extragerea informațiilor (NER, RE), răspuns la întrebări (QA), sumarizare abstractivă. Bioinformatică: AlphaFold 2 — predicția structurii proteinelor.	2	Prelegere interactivă	
Tema 14: Etica avansată în ML și recapitulare. Fairness în ML: metrici de echitate (demographic parity, equalized odds), compromisul fairness-acuratețe. Atacuri adversariale: FGSM, PGD — robustețea modelelor. Confidențialitatea datelor: federated learning, differential privacy. Provocările LLM-urilor: hallucination, alignment. Recapitulare generală.	2	Prelegere, recapitulare	

Bibliografie obligatorie

1. Goodfellow, I., Bengio, Y., Courville, A., Deep Learning, MIT Press, 2016. Disponibil gratuit la: <https://www.deeplearningbook.org/>
2. Geron, A., Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 3rd ed., O'Reilly Media, 2022.
3. Prince, S.J.D., Understanding Deep Learning, MIT Press, 2023. Disponibil gratuit la: <https://udlbook.github.io/udlbook/>

Bibliografie complementară

4. Sutton, R.S., Barto, A.G., Reinforcement Learning: An Introduction, 2nd ed., MIT Press, 2018. Disponibil gratuit la: <http://incompleteideas.net/book/the-book-2nd.html>
5. Hamilton, W.L., Graph Representation Learning, Morgan and Claypool, 2020. Disponibil gratuit la: https://www.cs.mcgill.ca/~wlh/grl_book/

Resurse software:

6. PyTorch Documentation: <https://pytorch.org/docs/>
7. Hugging Face Documentation: <https://huggingface.co/docs>
8. MLflow Documentation: <https://mlflow.org/docs/latest/>

6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Metode kernel și SVM. Antrenarea unui SVM cu kernel RBF pe un dataset nelinear separabil (scikit-learn). Vizualizarea hiperplanului de separare și a vectorilor suport. Optimizarea C și gamma cu GridSearchCV. Compararea cu regresia logistică.	2	Lucrare practică Python, scikit-learn	
Lucrarea L2: XGBoost și LightGBM pe date tabulare. Antrenarea XGBoost și LightGBM pe un dataset Kaggle (ex. House Prices sau Titanic). Compararea cu Random Forest. Optimizarea hiperparametrilor cu Optuna. Analiza importanței caracteristicilor.	2	Lucrare practică Python, XGBoost	
Lucrarea L3: Vision Transformer (ViT) cu fine-tuning. Fine-tuning al unui model ViT pretrained (Hugging Face) pe un dataset de imagini cu 5 clase (Google Colab). Compararea cu ResNet fine-tuned: acuratețe, timp de antrenare, numărul de parametri.	2	Lucrare practică PyTorch, Colab	
Lucrarea L4: BERT pentru clasificarea textului. Fine-tuning al unui model BERT (bert-base-uncased) pe un dataset de clasificare a sentimentelor (SST-2 sau un dataset românesc). Utilizarea Hugging Face	2	Lucrare practică Hugging Face	

Trainer API. Compararea cu TF-IDF + Logistic Regression.			
Lucrarea L5: Generarea de imagini cu GAN. Implementarea unui DCGAN pentru generarea de cifre MNIST cu PyTorch. Antrenarea discriminatorului și generatorului alternativ. Vizualizarea imaginilor generate la fiecare epocă. Identificarea semnelor de mode collapse.	2	Lucrare practică PyTorch	
Lucrarea L6: Graph Neural Network cu PyTorch Geometric. Implementarea unui GCN pentru clasificarea nodurilor pe dataset-ul Cora (citări între articole). Compararea cu un clasificator MLP care ignoră structura grafului. Vizualizarea embedding-urilor cu t-SNE.	2	Lucrare practică PyG, Colab	
Lucrarea L7: Explicabilitate cu SHAP. Antrenarea unui model XGBoost pe un dataset tabular de risc de credit. Calcularea valorilor SHAP globale (importanța caracteristicilor) și locale (explicarea unei predicții individuale). Generarea unui waterfall plot și a unui beeswarm plot.	2	Lucrare practică Python, SHAP	
Lucrarea L8: MLflow — tracking și model registry. Instrumentarea unui experiment de antrenare PyTorch cu MLflow: logarea parametrilor, metricilor și artefactelor. Compararea a 3 rulări cu hiperparametri diferiți în UI-ul MLflow. Înregistrarea modelului cel mai bun în Model Registry.	2	Lucrare practică MLflow	
Lucrarea L9: Serving ML cu FastAPI și Docker. Exportarea unui model antrenat în ONNX. Construirea unui serviciu de inferență cu FastAPI care primește JSON și returnează predicția. Containerizarea cu Docker. Testarea cu curl și Postman.	2	Lucrare practică FastAPI, Docker	
Lucrarea L10: Monitorizarea drift-ului în producție. Simularea drift-ului de date: antrenarea unui model pe date istorice, generarea de date noi cu distribuție schimbată. Detectarea drift-ului cu Evidently: raportul de data drift și graficele de distribuție. Praguri de alertare.	2	Lucrare practică Python, Evidently	
Lucrarea L11: Antrenare distribuită cu PyTorch DDP. Antrenarea unui model ResNet pe CIFAR-10 cu DistributedDataParallel (DDP) pe 2 procese simulate. Compararea timpului de antrenare DDP vs. antrenare pe un singur GPU. Analiza speedup-ului și eficienței.	2	Lucrare practică PyTorch DDP	
Lucrarea L12: Fine-tuning eficient cu LoRA. Fine-tuning al unui model de limbaj mic (GPT-2 sau Llama-3.2-1B) cu LoRA (PEFT library) pe un dataset de instrucțiuni. Compararea numărului de parametri	2	Lucrare practică PEFT, Colab	

antrenabili LoRA vs. full fine-tuning. Evaluarea calitativă a răspunsurilor generate.			
Lucrarea L13: Atacuri adversariale și robustețe. Implementarea atacului FGSM pe un model CNN antrenat pe MNIST. Vizualizarea perturbațiilor adversariale. Antrenarea unui model cu adversarial training și compararea robusteții. Calcularea acurateței pe exemple adversariale.	2	Lucrare practică PyTorch	
Lucrarea L14: Audit etic al unui model ML. Antrenarea unui clasificator pe un dataset cu potențial bias (ex. Adult Income sau COMPAS recidivism). Calcularea metricilor de fairness (demographic parity difference, equalized odds) cu Fairlearn. Identificarea grupurilor dezavantajate și propunerea măsurilor de mitigare. Raport scris cu concluzii etice.	2	Lucrare practică Fairlearn, raport	
<p>Bibliografie obligatorie laborator</p> <ol style="list-style-type: none"> 1. PyTorch Documentation: https://pytorch.org/docs/ 2. Hugging Face Documentation: https://huggingface.co/docs 3. MLflow Documentation: https://mlflow.org/docs/latest/ <p>Bibliografie complementara laborator</p> <ol style="list-style-type: none"> 4. SHAP Documentation: https://shap.readthedocs.io/ 5. Fairlearn Documentation: https://fairlearn.org/main/user_guide/ <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. PyTorch Documentation: https://pytorch.org/docs/ 7. Hugging Face Documentation: https://huggingface.co/docs 8. MLflow Documentation: https://mlflow.org/docs/latest/ 9. SHAP Documentation: https://shap.readthedocs.io/ 			

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

<p>Față de disciplina Inteligență Artificială (TIDS403) care acoperă algoritmi clasici și fundamentele ML, această disciplină se concentrează pe tehnicile avansate de deep learning, MLOps și responsabilitatea în producție. Laboratoarele acoperă ciclul complet: antrenare avansată, explicabilitate, serving, monitorizare și audit etic.</p> <p>Auditul etic din L14 și metrici de fairness formează direct R17 — responsabilitatea față de GDPR și standardele etice — diferențiind un practician ML responsabil de unul care optimizează exclusiv acuratețea.</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 arhitecturilor avansate (ViT, Transformer, GNN, VAE, GAN) și a cazurilor de utilizare; cunoașterea principiilor MLOps (tracking, registry, serving, monitorizare); identificarea riscurilor de bias și a metricilor de fairness pentru un	Examen scris: analiza comparativă a două arhitecturi pentru o problemă dată, proiectarea unui pipeline MLOps minimal și identificarea unui risc etic cu metrici de fairness adecvate	60%

	scenariu dat; cunoașterea tehnicilor de eficiență (LoRA, quantization, DDP).		
8.5 Seminar/laborator	Calitatea implementărilor din laboratoare (modelele antrenate corect, metricile calculate și interpretate); calitatea raportului de audit etic din L14 (grupurile dezavantajate identificate, metricile de fairness calculate, măsuri de mitigare propuse).	Implementări predate pe Sakai după fiecare lucrare (50%) și raportul de audit etic din L14 cu prezentare orală (50%)	40%
8.6 Standard minim de performanță:			
Raportul de audit etic din L14 trebuie să includă cel puțin două metrici de fairness calculate corect și o propunere concretă de mitigare cu estimarea impactului asupra acurateții. Un raport care constată existența biasului fără a propune măsuri sau fără metrici cuantificabile nu îndeplinește standardul minim.			

Data completării	Semnătura titularului de curs	Semnătura titularului de seminar / laborator
16.03.2026	Conf. Dr. Beteringhe Adrian	Conf. Dr. Beteringhe Adrian
Data avizării în Consiliul Departamentului	Semnătura directorului DIT	
24.03.2026	Lect. dr. Antohe Valerian	
Data aprobării în Consiliul Facultății	Semnătura decanului FMDT	
07.04.2026	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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei	BIOINFORMATICĂ – TIDS417						
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	IV	1.6 Semestrul	1	1.7 Tipul de evaluare	E	1.8 Regimul disciplinei	OPT

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					7
Referate					5
Teme casă					
Proiect					
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	Inteligență Artificială (TIDS403); Matematici Aplicate (TIDS206) — probabilități, statistică; Baze de Date (TIDS305).
3.2 de competențe	Cunoașterea algoritmilor ML de bază (clasificare, clustering) și a bibliotecilor scikit-learn și BioPython; înțelegerea noțiunilor de bază de biologie moleculară: ADN, ARN, proteine, gene.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu videoprojector și acces la internet; platforma Sakai; recomandare: cont NCBI, Ensembl și Google Colab activ înainte de curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator cu calculatoare individuale sau Google Colab; Python cu BioPython, scikit-learn, PyTorch, scanpy, pydeSeq2; acces la bazele de date publice NCBI, UniProt, Ensembl; 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	C17. Cunoaște în profunzime principiile inteligenței artificiale — algoritmi de căutare, reprezentarea cunoștințelor, sisteme expert — și metodele de învățare automată (Machine Learning), cu conștientizarea critică a limitelor și implicațiilor etice ale acestora. C19. Posedă cunoștințe avansate de inteligență computațională integrată — rețele neuronale, algoritmi evolutivi, logică fuzzy — cu capacitatea de a evalua critic aplicabilitatea acestora în rezolvarea problemelor ingineresti complexe.
5.2 Aptitudini	A17. Proiectează și implementează sisteme de inteligență artificială și modele de învățare automată (Machine Learning), evaluând critic performanța acestora și adaptând hiperparametrii pentru optimizarea rezultatelor.
5.3 Responsabilitate și autonomie	R17. Acționează în mod responsabil și etic în toate activitățile profesionale din domeniul TI, respectând drepturile de proprietate intelectuală, legislația privind protecția datelor cu caracter personal (GDPR) și standardele de conduită profesională.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Introducere în bioinformatică. Definiția și ramurile bioinformaticii: genomică, proteo-mică, transcripto-mică, metabolo-mică. Datele biologice: tipurile (secvențe, structuri 3D, expresie genică), volumul și provocările. Bazele de date publice: NCBI, Ensembl, UniProt, PDB. Formate standard: FASTA, FASTQ, VCF, GFF.	2	Prelegere interactivă	
Tema 2: Alinierea secvențelor. Alinierea globală: algoritmul Needleman-Wunsch. Alinierea locală: algoritmul Smith-Waterman. Matricele de substituție: PAM, BLOSUM62. BLAST: algoritmul heurisc, parametri (E-value, identitate). Alinierea multiplă: ClustalW, MUSCLE, MAFFT.	2	Prelegere interactivă	
Tema 3: Genomică și asamblarea genomului. Secvențierea Next-Generation Sequencing (NGS): Illumina, Oxford Nanopore, PacBio. Asamblarea de novo: algoritmi bazați pe grafuri De Bruijn (Velvet, SPAdes). Asamblarea prin mapare pe referință: BWA, Bowtie2. Evaluarea calității asamblării: N50, QUAST.	2	Prelegere interactivă	
Tema 4: Adnotarea genomică. Predicția genelor: GeneMark, Augustus. Adnotarea funcțională: GO (Gene Ontology), KEGG Pathways. Adnotarea structurală vs. funcțională. Variante genetice: SNP-uri, indel-uri, VCF. Analiza variantelor: GATK, SnpEff. Baze de date de variante: dbSNP, ClinVar.	2	Prelegere interactivă	
Tema 5: Transcriptomică și analiza expresiei genice. ARN-mesager și expresia genică. RNA-Seq: de la citiri la matrice de expresie (STAR, HISAT2, featureCounts). Analiza expresiei diferențiale: DESeq2, edgeR —	2	Prelegere interactivă	

normalizarea, testul statistic, corectarea pentru comparații multiple (FDR). Vizualizarea: volcano plot, heatmap.			
Tema 6: Single-cell RNA sequencing. Tehnologia scRNA-seq: 10x Chromium, Drop-seq. Pipeline-ul de procesare: filtrarea celulelor de calitate scăzută, normalizarea, detectarea genelor înalt variabile. Reducerea dimensionalității: PCA, UMAP. Clustering-ul celulelor: Leiden, Louvain. Adnotarea tipurilor celulare.	2	Prelegere interactivă	
Tema 7: Proteo-mică computațională. Structura proteinelor: nivelele primar, secundar, terțiar, cuaternar. Predicția structurii secundare: PSIPRED. AlphaFold 2: arhitectura Evoformer, datele MSA, performanța. Baza de date AlphaFold DB. Docking molecular: AutoDock Vina — aplicații în descoperirea medicamentelor.	2	Prelegere interactivă	
Tema 8: Filogenetica computațională. Construcția arborilor filogenetici: distanță (UPGMA, Neighbor-Joining), parsimonie, bayesian (MrBayes). Modelele de evoluție: JC69, GTR. Bootstrapping: evaluarea robusteții. Vizualizarea arborilor: FigTree, iTOL. Filoge-neza SARS-CoV-2: studiu de caz.	2	Prelegere interactivă	
Tema 9: Machine Learning în genomică. Clasificarea secvențelor biologice cu ML: reprezentarea k-mer. Rețele neuronale convoluționale pentru ADN: DeepBind (predicția siturilor de legare a factorilor de transcripție). Modele de limbaj pentru proteine: ESM-2 (Meta). Predicția efectului variantelor genetice cu ML.	2	Prelegere interactivă	
Tema 10: Rețele biologice. Tipurile de rețele biologice: protein-protein interaction (PPI), rețele de reglare genică, rețele metabolice. Analiza topologiei: grade, clustering coefficient, hub-uri. Algoritmi de detecție a comunităților în rețele biologice. Cytoscape: vizualizarea și analiza. STRING DB: interacțiunile proteice.	2	Prelegere interactivă	
Tema 11: Epidemiologie computațională. Modelele compartimentale: SIR, SEIR — ecuațiile diferențiale, R0. Filogeneza moleculară în epidemiologie: trasarea transmiterii. BEAST2: analiza bayesiană a datelor temporale. Genomica pandemiei COVID-19: GISAID, Nextstrain. Rezistența la antibiotice: detectarea genelor AMR.	2	Prelegere interactivă	
Tema 12: Descoperirea medicamentelor computaționale. Ciclul de descoperire a medicamentelor. Virtual screening: docking molecular, farmacofor. QSAR (Quantitative Structure-Activity Relationship): modelarea	2	Prelegere interactivă	

cu ML. Rețelele neuronale pentru molecule: Graph Neural Networks pe grafuri moleculare (mpnn). Bazele de date: ChEMBL, DrugBank.			
Tema 13: Bioinformatica clinică și medicina de precizie. Medicina de precizie: diagnosticul genomic, terapiile țintite. Secvențierea exomului și a genomului întreg în clinică. Interpretarea variantelor clinice: ACMG guidelines. Datele de sănătate: HIPAA, GDPR în contextul datelor genomice. Biomarkeri moleculari: descoperire și validare.	2	Prelegere interactivă	
Tema 14: Tendințe și recapitulare. Biologia sintetică: proiectarea circuitelor genetice. Genomica spațială: Visium, MERFISH — expresia genică în context spațial. Nanopori și secvențierea în timp real. Integrarea multi-omics: genomică + transcriptomică + proteo-mică + metabolo-mică. Implicații etice: consimțământul în genomică, datele genetice și discriminarea. Recapitulare generală.	2	Prelegere, recapitulare	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> 1. Durbin, R., Eddy, S., Krogh, A., Mitchison, G., Biological Sequence Analysis, Cambridge University Press, 1998. 2. Compeau, P., Pevzner, P., Bioinformatics Algorithms: An Active Learning Approach, 3rd ed., Active Learning Publishers, 2018. Material parțial gratuit la: https://www.bioinformaticsalgorithms.org/ 3. BioPython Documentation: https://biopython.org/wiki/Documentation <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Alberts, B. et al., Molecular Biology of the Cell, 7th ed., Norton, 2022. 5. NCBI Resources — baze de date bioinformatic: https://www.ncbi.nlm.nih.gov/ <p>Resurse software:</p> <ol style="list-style-type: none"> 6. BioPython Documentation: https://biopython.org/wiki/Documentation 7. NCBI BLAST: https://blast.ncbi.nlm.nih.gov/ 8. Ensembl Genome Browser: https://www.ensembl.org/ 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: BioPython — manipularea secvențelor. Descărcarea și parsarea unui genom bacterian din NCBI cu BioPython (Entrez). Căutarea motivelor cu expresii regulate. Calcularea compoziției nucleotidice și a conținutului GC. Transcripția și translația in silico.	2	Lucrare practică Python, BioPython	
Lucrarea L2: Alinierea secvențelor cu BLAST și BioPython. Rularea BLAST local cu BioPython pe un set de secvențe proteice. Parsarea rezultatelor XML: extragerea E-value, identității, lungimii alinierii. Alinierea multiplă cu MUSCLE prin BioPython. Vizualizarea alinierii.	2	Lucrare practică BioPython, BLAST	
Lucrarea L3: Controlul calității datelor NGS. Analiza calității unui fișier FASTQ real cu FastQC (sau o implementare Python echivalentă). Interpretarea scorurilor Phred, distribuției lungimilor citirilor, conținutului adaptor. Trimming-ul citirilor de calitate slabă cu Trimmomatic sau cutadapt.	2	Lucrare practică Python, FastQC	
Lucrarea L4: Maparea citirilor pe genom de referință. Maparea citirilor RNA-Seq pe	2	Lucrare practică Python, HISAT2	

genomul de referință al unui organism model (ex. E. coli sau D. melanogaster) cu HISAT2 sau STAR (prin subprocess Python sau Galaxy). Generarea fișierului BAM. Calcularea statisticilor de mapare.			
Lucrarea L5: Analiza expresiei diferențiale. Cuantificarea transcriptelor cu featureCounts pe fișierele BAM din L4. Analiza expresiei diferențiale cu pydeseq2 în Python: normalizarea, identificarea genelor semnificative ($FDR < 0.05$, $ \log_2FC > 1$). Generarea unui volcano plot și a unui heatmap.	2	Lucrare practică Python, pydeseq2	
Lucrarea L6: Analiza scRNA-seq cu scanpy. Analiza unui dataset scRNA-seq public (ex. PBMC 3k de la 10x Genomics) cu scanpy: filtrarea celulelor de calitate slabă, normalizarea, detecția genelor înalt variabile, PCA, UMAP, clustering Leiden. Adnotarea manuală a clusterelor pe baza markerilor genici.	2	Lucrare practică Python, scanpy	
Lucrarea L7: Predicția structurii proteinelor cu AlphaFold. Rularea AlphaFold 2 pe ColabFold pentru o proteină de interes (sub 400 aminoacizi). Vizualizarea structurii 3D cu py3Dmol sau ChimeraX. Interpretarea scorului pLDDT. Compararea cu structura experimentală din PDB (dacă există).	2	Lucrare practică ColabFold, py3Dmol	
Lucrarea L8: Filogenetică computațională. Construirea unui arbore filogenetic pentru un set de secvențe proteice (ex. citocromul C la 15 specii) cu BioPython: alinierea MUSCLE, calculul distanțelor, construcția NJ. Exportul în format Newick și vizualizarea cu ETE3 sau Phylo din BioPython.	2	Lucrare practică BioPython, ETE3	
Lucrarea L9: ML pentru clasificarea secvențelor genomice. Reprezentarea secvențelor ADN ca k-meri ($k=4$, $k=6$). Antrenarea unui clasificator Random Forest pentru distincția promotori vs. regiuni non-promotor. Compararea cu un model CNN simplu (PyTorch) pe aceeași problemă. Evaluarea cu ROC-AUC.	2	Lucrare practică Python, PyTorch	
Lucrarea L10: Analiza rețelelor de interacțiune proteică. Descărcarea rețelei PPI pentru un organism (ex. S. cerevisiae) din STRING DB cu Python (API). Construirea grafului cu NetworkX. Calcularea centralității (degree, betweenness). Identificarea hub-urilor. Vizualizarea cu Matplotlib sau Cytoscape.	2	Lucrare practică Python, NetworkX	
Lucrarea L11: Docking molecular cu AutoDock Vina. Pregătirea unui ligand și a unei ținte proteice (din PDB) pentru docking cu AutoDock Vina (prin subprocess Python). Rularea docking-ului. Interpretarea energiei	2	Lucrare practică AutoDock Vina, Python	

de legare (kcal/mol). Vizualizarea complexului ligand-receptor cu py3Dmol.			
Lucrarea L12: Genomica epidemiologică. Descărcarea unui set de secvențe SARS-CoV-2 din GISAID (sau dataset public similar). Alinierea cu MAFFT. Construcția arborelui filogenetic cu IQ-TREE sau FastTree. Vizualizarea pe Nextstrain sau FigTree. Identificarea clusterelor de transmitere.	2	Lucrare practică MAFFT, IQ-TREE	
Lucrarea L13: QSAR și predicția activității moleculare. Descărcarea unui dataset de molecule cu activitate biologică măsurată din ChEMBL cu Python (chembl_webresource_client). Calcularea descriptorilor moleculari cu RDKit. Antrenarea unui model Random Forest pentru predicția IC50. Evaluarea și interpretarea modelului.	2	Lucrare practică Python, RDKit	
Lucrarea L14: Proiect de analiză bioinformatică și prezentare. Realizarea unui pipeline bioinformatic complet pe un dataset ales (expresie diferențială, clasificare secvențe, rețea PPI sau QSAR): preprocesare, analiză, vizualizare, interpretare biologică. Redactarea unui raport tehnic de 3-4 pagini. Prezentarea rezultatelor cu accent pe interpretarea biologică a concluziilor.	2	Proiect individual, prezentare	
	<p>Bibliografie obligatorie laborator</p> <p>1. BioPython Documentation: https://biopython.org/wiki/Documentation</p> <p>2. scanpy Documentation: https://scanpy.readthedocs.io/</p> <p>3. NCBI Resources: https://www.ncbi.nlm.nih.gov/</p> <p>Bibliografie complementara laborator</p> <p>4. ColabFold — AlphaFold 2 gratuit: https://colab.research.google.com/github/sokrypton/ColabFold/</p> <p>5. RDKit Documentation: https://www.rdkit.org/docs/</p> <p>Resurse online gratuite recomandate</p> <p>6. BioPython Documentation: https://biopython.org/wiki/Documentation</p> <p>7. scanpy Documentation: https://scanpy.readthedocs.io/</p> <p>8. NCBI BLAST: https://blast.ncbi.nlm.nih.gov/</p> <p>9. ColabFold: https://colab.research.google.com/github/sokrypton/ColabFold/</p>		

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

Bioinformatica aplică direct competențele ML (C17, C19, A17) la date biologice reale: secvențe genomice, date de expresie genică, structuri proteice și rețele de interacțiune. Disciplina formează capacitatea de a transpune metodele ML dintr-un domeniu general în probleme specifice, cu înțelegerea profundă a datelor de intrare.

Componentele etice — confidențialitatea datelor genomice, consimțământul în cercetare și implicațiile medicinei de precizie — formează direct R17, plasând responsabilitatea față de GDPR și drepturile persoanelor în contextul concret al datelor de sănătate.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
----------------	--------------------------	------------------------	-----------------------------

8.4 Curs	Cunoașterea algoritmilor de aliniere (Needleman-Wunsch, Smith-Waterman, BLAST) și a formatelor de date biologice; cunoașterea pipeline-ului RNA-Seq (de la citiri la expresie diferențială); cunoașterea structurii proteinelor și a principiului AlphaFold; identificarea implicațiilor etice ale datelor genomice medicale.	Examen scris: interpretarea unui rezultat BLAST dat, proiectarea unui pipeline RNA-Seq minimal pentru o problemă descrisă și identificarea a două riscuri etice specifice datelor genomice clinice	50%
8.5 Seminar/laborator	Corectitudinea implementărilor din laborator (pipeline-urile funcționează pe datele date); calitatea raportului din L14: interpretarea biologică a rezultatelor, nu doar descrierea tehnică a pașilor.	Rapoarte de laborator predate pe Sakai după fiecare lucrare (50%) și raportul de analiză din L14 cu prezentare orală (50%)	50%
8.6 Standard minim de performanță:			
Raportul din L14 trebuie să includă o concluzie biologică — nu una tehnică — fundamentată pe rezultatele analizei: o ipoteză despre funcția biologică, boala sau mecanismul molecular studiat, justificată prin datele obținute. Un raport care descrie exclusiv pașii tehnici fără interpretare biologică nu îndeplinește standardul minim.			

Data completării	Semnătura titularului de curs	Semnătura titularului de seminar / laborator
16.03.2026	Conf. Dr. Beteringhe Adrian	Conf. Dr. Beteringhe Adrian
Data avizării în Consiliul Departamentului	Semnătura directorului DIT	
24.03.2026	Lect. dr. Antohe Valerian	
Data aprobării în Consiliul Facultății	Semnătura decanului FMDT	
07.04.2026	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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei	CLOUD COMPUTING ȘI ARHITECTURI DISTRIBUITE – TIDFAC422						
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	IV	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	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					69
Documentare					22
Studiu individual					22
Referate					22
Teme casă					
Proiect					
Alte activități (inclusive colocviu, examen)					3
2.9 Total ore pe semestru					125
2.10 Numărul de credite					5

3. Precondiții (acolo unde este cazul)

3.1 de curriculum	Cloud Computing (TIDFAC324); Rețele de Calculatoare (TIDS301); Sisteme Avansate de Baze de Date; Ingineria Programelor (TIDS308) — Git, CI/CD.
3.2 de competențe	Cunoașterea teoremei CAP și a modelelor de consistență; familiarizarea cu Docker și Kubernetes la nivel mediu; înțelegerea arhitecturii microserviciilor; experiența cu un message broker (Kafka sau RabbitMQ) la nivel introductiv.

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 sau GCP Free Tier și Docker Desktop instalat înainte de curs.
4.2 de desfășurare a seminarului/laboratorului	Laborator cu calculatoare individuale; Docker Desktop, kubectl, Minikube; Python cu kafka-python, boto3, grpc; acces la GitHub și la un registry de containere; 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	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. A34. Proiectează și implementează pipeline-uri de date pentru arhitecturi distribuite și cloud, utilizând tehnologii de streaming (Kafka, Spark) și asigurând consistența și disponibilitatea datelor în sisteme cu cerințe înalte de scalabilitate.
5.3 Responsabilitate și autonomie	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: Sisteme distribuite — fundamente avansate. Modelele de consistență: linearizabilitate, consistența secvențială, consistența eventuală, CRDT-uri. Teorema CAP: demonstrația imposibilității, sistemele CP vs. AP. Teorema PACELC: extensia cu latența. Algoritmul Raft: liderul, jurnalul replicat, votul, scenariile de eșec.	2	Prelegere interactivă	
Tema 2: Arhitecturi cloud avansate. Modele de deployment: public, privat, hibrid, multi-cloud. Serviciile managed vs. self-managed: compromisurile. Arhitecturi serverless: AWS Lambda, Google Cloud Functions — cold start, limitele, tarifare per-execuție. Edge computing: CloudFront, Wavelength. FinOps: modelele de cost, optimizarea cheltuielilor cloud.	2	Prelegere interactivă	
Tema 3: Microservicii — arhitectura și pattern-urile. Decompoziția în microservicii: by business capability, by subdomain (DDD). Comunicarea sincronă: REST, gRPC. Comunicarea asincronă: event-driven, message queues. Pattern-urile microserviciilor: API Gateway, Circuit Breaker, Saga (coreografie vs. orchestrare), CQRS, Event Sourcing.	2	Prelegere interactivă	
Tema 4: Kubernetes avansat. Operatori Kubernetes: CRD-uri, controller loop. Custom Resource Definitions: definirea resurselor proprii. Kubernetes networking: CNI plugins (Calico, Cilium), Network Policies, Service Mesh. Kubernetes security: RBAC, Pod Security Standards, Secrets management. Multi-cluster: Cluster API, ArgoCD ApplicationSets.	2	Prelegere interactivă	

<p>Tema 5: Apache Kafka avansat. Arhitectura Kafka: brokeri, topicuri, partiții, offset-uri, grupuri de consumatori. Replicarea și durabilitatea: ISR, acks=all, min.insync.replicas. Kafka Streams: procesarea stateful, KTable, KStream, ferestre de timp. Kafka Connect: conectorii source și sink. Schema Registry: Avro, Protobuf, compatibilitatea schemelor.</p>	2	Prelegere interactivă	
<p>Tema 6: Procesarea distribuită a datelor. Apache Spark: RDD, DataFrame, lazy evaluation, DAG-ul de execuție. Spark Structured Streaming: micro-batch vs. continuous processing, watermarking. Apache Flink: procesarea true streaming, starea gestionată, checkpointing. Compararea Spark vs. Flink. Delta Lake: ACID pe data lake.</p>	2	Prelegere interactivă	
<p>Tema 7: Service Mesh și observabilitatea distribuită. Istio: arhitectura (control plane, data plane, Envoy sidecar), traffic management, mutual TLS. Tracing distribuit: OpenTelemetry, Jaeger — propagarea contextului, span-urile. Logging centralizat: ELK Stack, Loki. SRE: SLI, SLO, SLA, error budgets.</p>	2	Prelegere interactivă	
<p>Tema 8: Toleranța la defecte și reziliența. Failure modes în sisteme distribuite: crash, omission, Byzantine. Bulkhead pattern: izolarea resurselor. Retry cu exponential backoff și jitter. Timeout-urile: cascading failures. Chaos Engineering: principiile, Chaos Monkey, experimentele controlate. Disaster Recovery: RTO, RPO, strategiile.</p>	2	Prelegere interactivă	
<p>Tema 9: Consistența datelor în sisteme distribuite. Two-Phase Commit (2PC): fazele, eșecurile. Saga pattern: tranzacțiile compensatorii. Idempotența: implementarea în API-uri și consumatori de mesaje. Distributed locks: Redis RedLock, ZooKeeper. Optimistic vs. pessimistic locking la scară. CRDT-uri: tipurile și aplicațiile.</p>	2	Prelegere interactivă	
<p>Tema 10: Arhitecturi cloud-native pentru AI/ML. MLOps în cloud: SageMaker, Vertex AI, Azure ML. Inferența la scară: Triton Inference Server, KServe în Kubernetes. Vector databases în cloud: Pinecone, pgvector pe RDS. RAG (Retrieval-Augmented Generation): arhitectura, pipeline-ul de date. GPU-uri în cloud: tipurile de instanțe, costul.</p>	2	Prelegere interactivă	
<p>Tema 11: Securitatea arhitecturilor distribuite. Zero Trust Architecture: principiile, implementarea în cloud. IAM: federated identity, OIDC, SPIFFE/SPIRE pentru workload identity. Secrets management la scară: HashiCorp Vault, AWS Secrets Manager, rotația automată. Network security</p>	2	Prelegere interactivă	

în cloud: VPC peering, PrivateLink. Compliance: SOC 2, ISO 27001.			
Tema 12: Infrastructure as Code și GitOps avansat. Terraform la scară: module-uri, workspace-uri, Terragrunt, remote state. Pulumi: IaC cu limbaje de programare general-purpose. Crossplane: provisionarea infrastructurii cloud din Kubernetes. GitOps multi-cluster: ArgoCD ApplicationSets. Policy as Code: OPA, Kyverno.	2	Prelegere interactivă	
Tema 13: Optimizarea costurilor și performanței cloud. Reserved Instances vs. Spot Instances vs. Savings Plans. Right-sizing: instrumentele AWS Compute Optimizer, Azure Advisor. Data transfer costs: optimizarea prin CDN și regiuni. Performance tuning: profilarea aplicațiilor distribuite, distribuția latenței (p50/p95/p99). Observabilitatea pentru cost.	2	Prelegere interactivă	
Tema 14: Tendințe și recapitulare. WebAssembly în cloud: WASI, Spin — alternativă la containere. eBPF: observabilitate și securitate la nivel de kernel, Cilium, Tetragon. Confidential computing: AWS Nitro Enclaves. Quantum-safe cloud: migrarea la algoritmi post-cuantici. Platform Engineering: IDP-uri, Backstage. Recapitulare generală.	2	Prelegere, recapitulare	
	<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. Tanenbaum, A.S., Van Steen, M., Distributed Systems: Principles and Paradigms, 3rd ed., Pearson, 2017. Disponibil gratuit la: https://www.distributed-systems.net/ <p>Bibliografie complementară</p> <ol style="list-style-type: none"> 4. Richardson, C., Microservices Patterns, Manning Publications, 2018. 5. AWS Documentation: https://docs.aws.amazon.com/ <p>Resurse software:</p> <ol style="list-style-type: none"> 6. Kubernetes Documentation: https://kubernetes.io/docs/ 7. Apache Kafka Documentation: https://kafka.apache.org/documentation/ 8. AWS Documentation: https://docs.aws.amazon.com/ 		
6.2 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Algoritm Raft — simulare. Simularea algoritmului Raft cu un simulator vizual (raft.github.io sau implementare Python cu 5 noduri). Observarea alegerilor de lider, a replicării jurnalului și a comportamentului la eșecul unui nod. Analiza comportamentului CP la pierderea quorum-ului.	2	Lucrare practică Python, simulator	
Lucrarea L2: Microservicii cu Docker Compose. Implementarea unei aplicații cu 3 microservicii (users, orders, payments) cu comunicare REST și asincronă prin Redis Pub/Sub. Configurarea cu Docker Compose. Implementarea Circuit Breaker cu biblioteca tenacity (Python): testarea comportamentului la eșecul unui serviciu.	2	Lucrare practică Docker Compose	

Lucrarea L3: gRPC pentru comunicarea inter-servicii. Definierea unui contract de serviciu cu Protocol Buffers. Implementarea unui server și client gRPC în Python. Compararea latenței și dimensiunii mesajelor gRPC vs. REST pentru același endpoint. Implementarea gRPC streaming bidirecțional.	2	Lucrare practică Python, gRPC	
Lucrarea L4: Kubernetes — CRD și operatori simpli. Crearea unui Custom Resource Definition (CRD) simplu în Kubernetes (Minikube). Implementarea unui controller Python minimal cu kopf care reconciliază resursa custom. Testarea creării, modificării și ștergerii resursei și observarea acțiunilor controller-ului.	2	Lucrare practică Kubernetes, Python	
Lucrarea L5: Apache Kafka — producători și consumatori. Configurarea unui cluster Kafka cu Docker Compose (3 brokeri, ZooKeeper sau KRaft). Implementarea unui producer și a unui consumer Python cu kafka-python. Testarea durabilității: oprirea unui broker și verificarea că mesajele nu se pierd (acks=all, replication factor=3).	2	Lucrare practică Kafka, Python	
Lucrarea L6: Kafka Streams — procesarea stateful. Implementarea unui pipeline de procesare cu Faust (Python): consumarea unui topic de evenimente de vânzări, agregarea pe fereastră de timp de 1 minut (vânzări totale per produs), publicarea rezultatelor pe un topic de ieșire. Vizualizarea în timp real.	2	Lucrare practică Faust, Python	
Lucrarea L7: Saga pattern — tranzacție distribuită. Implementarea unui flux de comandă cu Saga coreografic folosind Kafka: serviciul Orders publică un eveniment, serviciile Payment și Inventory îl consumă și publică răspunsuri. Implementarea tranzacției compensatorii la eșecul plății: anularea comenzii.	2	Lucrare practică Kafka, Python	
Lucrarea L8: Tracing distribuit cu OpenTelemetry. Instrumentarea aplicației cu microservicii din L2 cu OpenTelemetry SDK (Python). Configurarea unui exporter Jaeger local (Docker). Vizualizarea trace-urilor end-to-end: identificarea serviciului cu latența cea mai mare și a cauzei blocajului.	2	Lucrare practică OpenTelemetry, Jaeger	
Lucrarea L9: Chaos Engineering. Aplicarea experimentelor de chaos pe clusterul Minikube cu Chaos Mesh sau prin oprirea manuală a pod-urilor. Testarea comportamentului aplicației la: eșecul unui pod, latență injectată (100ms) pe o rută, pierdere de pachete (20%). Documentarea comportamentului observat și a măsurilor de mitigare.	2	Lucrare practică Kubernetes, Chaos Mesh	
Lucrarea L10: Terraform — provisionarea infrastructurii cloud. Scrierea unui modul	2	Lucrare practică Terraform, AWS	

Terraform care provisionează pe AWS Free Tier: un VPC cu subrețele publice/private, o instanță EC2, un Security Group și un S3 bucket. Utilizarea remote state (S3 backend). Demonstrarea terraform plan, apply, destroy.			
Lucrarea L11: Securitate cloud — IAM și secrets management. Configurarea unui utilizator IAM cu permisiuni minime (least privilege) pe AWS pentru o aplicație specifică. Stocarea secretelor în AWS Secrets Manager și accesul din Python cu boto3. Configurarea rotației automate a secretului. Auditarea accesului cu CloudTrail.	2	Lucrare practică AWS, Python	
Lucrarea L12: GitOps cu ArgoCD. Instalarea ArgoCD în Minikube. Crearea unui repository GitHub cu manifestele Kubernetes ale aplicației din L2. Configurarea unei ArgoCD Application cu sync automat. Demonstrarea reconcilierii: modificarea manuală a unui Deployment și observarea restaurării automate din Git.	2	Lucrare practică ArgoCD, Kubernetes	
Lucrarea L13: Optimizarea costurilor cloud. Analiza costurilor unui cont AWS Free Tier cu Cost Explorer. Calcularea economiilor estimate prin Reserved Instances vs. On-Demand pentru un scenariu dat cu AWS Pricing Calculator. Identificarea a 3 resurse neutilizate cu Trusted Advisor. Proiectarea unei strategii de cost pentru o arhitectură de microservicii.	2	Lucrare practică AWS Cost Explorer	
Lucrarea L14: Proiect integrat — arhitectură distribuită completă. Proiectarea și demonstrarea unui sistem distribuit cu: 3 microservicii containerizate în Kubernetes, comunicare asincronă prin Kafka, Saga pentru o tranzacție critică, tracing distribuit cu Jaeger și infrastructură provisionată cu Terraform. Raport tehnic cu justificarea deciziilor CAP/PACELC și prezentare orală.	2	Proiect individual, prezentare	
	<p>Bibliografie obligatorie laborator</p> <ol style="list-style-type: none"> 1. Kubernetes Documentation: https://kubernetes.io/docs/ 2. Apache Kafka Documentation: https://kafka.apache.org/documentation/ 3. AWS Documentation: https://docs.aws.amazon.com/ <p>Bibliografie complementara laborator</p> <ol style="list-style-type: none"> 4. ArgoCD Documentation: https://argo-cd.readthedocs.io/ 5. Terraform Documentation: https://developer.hashicorp.com/terraform/docs <p>Resurse online gratuite recomandate</p> <ol style="list-style-type: none"> 6. Kubernetes Documentation: https://kubernetes.io/docs/ 7. Apache Kafka Documentation: https://kafka.apache.org/documentation/ 8. ArgoCD Documentation: https://argo-cd.readthedocs.io/ 9. Terraform Documentation: https://developer.hashicorp.com/terraform/docs 		

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

Disciplina aprofundează competențele din Cloud Computing (An III) cu arhitecturi distribuite avansate: consensul distribuit (Raft), pattern-urile microserviciilor (Saga, CQRS, Event Sourcing), procesarea streaming cu Kafka și Flink și observabilitatea cu OpenTelemetry. Laboratoarele sunt cumulative — proiectul integrat din L14 combină componentele construite în L1-L13.

Justificarea deciziilor de design CAP/PACELC la prezentarea finală formează R34: asumarea responsabilității pentru compromisurile consistență-disponibilitate-toleranță la partiții ale arhitecturii alese, cu conștientizarea riscurilor operaționale în producție.

8. Evaluare

Tip activitate	8.1 Criterii de evaluare	8.2 Metode de evaluare	8.3 Pondere din nota finală
8.4 Curs	Cunoașterea algoritmului Raft și a garantărilor de consistență; justificarea clasificării CAP a unui sistem dat; proiectarea unui pipeline Kafka pentru o cerință de procesare specificată; alegerea pattern-ului potrivit (Saga, CQRS, Event Sourcing) pentru un scenariu de microservicii descris.	Examen scris: clasificarea CAP a unui sistem descris, proiectarea unui pipeline distribuit pentru o cerință dată și alegerea pattern-ului de reziliență adecvat unui scenariu de eșec specificat	40%
8.5 Seminar/laborator	Funcționalitatea proiectului integrat din L14: microserviciile comunică asincron prin Kafka, Saga compensează corect la eșec, trace-urile sunt vizibile în Jaeger, infrastructura Terraform se provisionează fără erori; calitatea raportului tehnic cu justificarea deciziilor CAP/PACELC.	Implementări predate pe Sakai după fiecare lucrare (50%) și proiect integrat cu prezentare orală la L14 (50%)	60%
8.6 Standard minim de performanță:			
Demonstrarea la L14 a unui sistem cu cel puțin un topic Kafka funcțional, un pattern Saga cu tranzacție compensatorie demonstrabilă și trace-uri vizibile în Jaeger pentru un flux end-to-end. Un sistem fără comunicare asincronă sau fără observabilitate nu îndeplinește standardul minim, indiferent de calitatea UI-ului.			

Data completării 16.03.2026	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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei		MANAGEMENTUL CLASEI DE ELEVI – TIDFAC424					
1.2 Titularul activităților de curs		Lect. Dr. Popa Doinița					
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator		Lect. Dr. Popa Doinița					
1.5 Anul de studiu	IV	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	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					47
Documentare					15
Studiu individual					15
Referate					15
Teme casă					
Proiect					
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	Pedagogie I, Pedagogie II, Psihologia educației, Didactica specializării (informatică).
3.2 de competențe	Cunoștințe de bază de psihopedagogie (teorii ale învățării, comunicare educațională); capacitatea de a proiecta o secvență didactică; familiaritate cu specificul predării disciplinelor informatice.

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	Sală de curs cu videoproiector și acces la internet; platforma Sakai; recomandare: studii de caz și înregistrări video pentru analiza situațiilor de management al clasei.
4.2 de desfășurare a seminarului/laboratorului	Sală de seminar cu mobilier modular pentru lucru în grup; flipchart, markere, fișe de observație; acces la resurse video pentru jocuri de rol și simulări.

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

5.1 Cunoștințe	C36. Cunoaște principiile didacticii specializării și ale managementului clasei, inclusiv strategii de evaluare formativă și summativă, adaptate predării disciplinelor tehnice din învățământul preuniversitar și universitar.
5.2 Aptitudini	A40. Gestionează clasa de elevi în contexte de educație tehnică, aplicând strategii de motivare, rezolvare a conflictelor și diferențiere a instruirii, asigurând un climat favorabil învățării disciplinelor informatice.
5.3 Responsabilitate și autonomie	R24. Contribuie în mod responsabil la transferul de cunoștințe tehnice specializate către beneficiari, colegi și parteneri, adaptând nivelul de detaliu tehnic al comunicării la profilul și nevoile interlocutorilor. R40. Gestionează autonom clasa de elevi în activitățile practice de informatică, asumând responsabilitatea pentru securitatea digitală a mediului de lucru al elevilor, respectarea regulamentelor școlare și prevenirea comportamentelor inadecvate în mediul online.

6. Conținuturi

6.1 Curs	Nr. ore	Metode de predare	Observații
Tema 1: Fundamentele managementului clasei. Definirea conceptului. Funcțiile managerului-profesor: planificare, organizare, coordonare, evaluare. Stiluri de management al clasei: autoritar, democratic, permisiv. Particularități în predarea disciplinelor informatice (laborator, lucru în echipă pe proiecte).	2	Prelegere interactivă, dezbateri	
Tema 2: Climatul psihosocial al clasei și relația profesor-elev. Dimensiunile climatului educațional: relațional, cognitiv, motivațional. Comunicarea didactică: verbală, nonverbală, paraverbală. Empatia, ascultarea activă. Construirea încrederii în orele de informatică: feedback constructiv pe cod, evitarea umilirii la public.	2	Prelegere interactivă, dezbateri	
Tema 3: Motivarea elevilor și diferențierea instruirii. Teorii motivaționale: Maslow, autodeterminarea (Deci & Ryan). Motivația intrinsecă vs. extrinsecă. Diferențierea pe niveluri (începători-avansați la programare). Strategii: gamification, coding challenges, mentorat între elevi, proiecte cu grad variabil de dificultate.	2	Prelegere interactivă, dezbateri	
Tema 4: Disciplina, regulile și prevenția comportamentelor inadecvate. Stabilirea regulilor împreună cu elevii. Modelul lui Kounin: cu-itate (withitness), tranziții line, suprapunere. Tehnici de prevenție: monitorizarea continuă, semnale nonverbale. Specific informatic: prevenția distragerii (jocuri, rețele sociale) în timpul orelor de laborator.	2	Prelegere interactivă, dezbateri	
Tema 5: Gestionarea conflictelor și a comportamentelor disruptive. Tipuri de conflicte: elev-elev, elev-profesor, grup-grup. Etapele rezolvării conflictelor (Glasser, Gordon). Tehnici de mediere și negociere. Bullying și cyberbullying — recunoaștere și intervenție. Protocol de raportare către consilier și conducerea școlii.	2	Prelegere interactivă, dezbateri	
Tema 6: Securitatea digitală și etica în mediul online școlar. Regulamentul de utilizare a laboratorului de informatică. Politici de parolă,	2	Prelegere interactivă, dezbateri	

sesiuni multiple, protejarea datelor elevilor (GDPR școlar). Filtrarea conținutului, monitorizarea activității. Educație pentru cetățenie digitală: dezinformare, drepturi de autor, comportament etic în comunitățile online.			
Tema 7: Evaluarea formativă și summativă. Recapitulare. Tipuri de evaluare: diagnostică, formativă, summativă. Instrumente specifice informaticii: portofolii de cod, evaluare prin peer review, examene practice pe calculator, grile de evaluare a proiectelor. Feedback eficient. Recapitulare integrativă a temelor și pregătirea pentru colocviu.	2	Prelegere interactivă, dezbateri	
	<p>Bibliografie obligatorie</p> <ol style="list-style-type: none"> Iucu, R.B., Managementul clasei de elevi. Aplicații pentru gestionarea situațiilor de criză educațională, Polirom, 2006. Stan, E., Managementul clasei, ed. a 2-a, Aramis, 2009. Pânișoară, I.O., Comunicarea eficientă, ed. a 4-a, Polirom, 2015. <p>Bibliografie complementară</p> <ol style="list-style-type: none"> Cucoș, C., Pedagogie, ed. a 3-a, Polirom, 2014. Cerghit, I., Metode de învățământ, ed. a 4-a, Polirom, 2018. Marzano, R.J., Marzano, J.S., Pickering, D.J., Classroom Management That Works, ASCD, 2003. <p>Resurse software:</p> <ol style="list-style-type: none"> Edutopia – resurse pentru managementul clasei: https://www.edutopia.org/topic/classroom-management ISE (Institutul de Științe ale Educației) – materiale didactice: https://ise.ro/ Salvați Copiii – ghiduri anti-bullying: https://www.salvaticopiii.ro/ 		
6.2 Seminar	Nr. ore	Metode de predare	Observații
Seminarul S1: Studiu de caz pe stiluri de management al clasei. Analiza a trei înregistrări video cu profesori de informatică din contexte diferite. Identificarea stilului dominant, a punctelor forte și a punctelor slabe. Discuție în grup și formularea de recomandări pentru fiecare cadru didactic observat.	2	Studiu de caz, joc de rol, dezbateri	
Seminarul S2: Joc de rol — comunicare didactică în laboratorul de informatică. Studenții simulează situații de predare a unei lecții de programare: explicarea unui concept dificil, oferirea de feedback la cod, gestionarea unei întrebări neașteptate. Analiza componentelor verbale și nonverbale ale comunicării.	2	Studiu de caz, joc de rol, dezbateri	
Seminarul S3: Proiectarea unei strategii de motivare pentru o clasă de informatică. Pe grupe, studenții elaborează o strategie de motivare pentru o clasă fictivă cu profil eterogen (3 elevi pasionați, 5 medii, 4 slabi). Includ elemente de gamification, mentorat și diferențiere. Prezentare și feedback încrucișat.	2	Studiu de caz, joc de rol, dezbateri	
Seminarul S4: Elaborarea unui regulament al laboratorului de informatică. Pe grupe, studenții redactează un regulament concret pentru o clasă a IX-a, cu reguli pentru utilizarea calculatoarelor, accesul la internet, lucrul în echipă, predarea proiectelor. Negocierea regulilor — simularea unei discuții cu elevii.	2	Studiu de caz, joc de rol, dezbateri	
Seminarul S5: Studiu de caz — gestionarea unui conflict în orele de informatică. Analiza unui scenariu: doi elevi se ceartă pentru contribuția la un proiect comun, unul acuză celălalt de copiere.	2	Studiu de caz, joc de rol, dezbateri	

Studentii propun pași concreți de mediere conform modelului Gordon. Joc de rol și debriefing.			
Seminarul S6: Detectarea și intervenția în cazuri de cyberbullying. Studii de caz reale (anonimizate): mesaje în grupul de WhatsApp al clasei, capturi de ecran, trolling pe platforma de e-learning. Studentii elaborează un plan de intervenție pas cu pas: documentare, comunicare cu părintii, raportare, prevenție.	2	Studiu de caz, joc de rol, dezbateri	
Seminarul S7: Construirea unui instrument de evaluare formativă pentru o lecție de informatică. Fiecare student elaborează o grilă de evaluare (rubric) pentru un proiect de programare la nivel de gimnaziu sau liceu, cu 4-5 criterii și niveluri de performanță descrise concret. Susținere orală și revizuire pe baza feedback-ului.	2	Studiu de caz, joc de rol, dezbateri	
<p>Bibliografie obligatorie seminar</p> <p>1. Iucu, R.B., Managementul clasei de elevi, Polirom, 2006. 2. Pânișoară, I.O., Profesorul de succes. 59 de principii de pedagogie practică, ed. a 2-a, Polirom, 2017. 3. Gordon, T., Profesorul eficient. Programul Gordon pentru îmbunătățirea relației cu elevii, Trei, 2011.</p> <p>Bibliografie complementara seminar</p> <p>4. Hattie, J., Învățarea vizibilă. Ghid pentru profesori, Trei, 2014. 5. Stan, E., Managementul clasei, Aramis, 2009.</p> <p>Resurse online gratuite recomandate</p> <p>6. Edutopia – Classroom Management Strategies: https://www.edutopia.org/topic/classroom-management 7. Common Sense Education – Digital Citizenship: https://www.commonsense.org/education/digital-citizenship 8. Ora de Net – siguranță online pentru școli: https://oradnet.salvaticopiii.ro/ 9. Edpuzzle, Kahoot, Mentimeter – instrumente pentru evaluare formativă</p>			

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

Disciplina formează direct competențele necesare profesorului de informatică: C36 — principiile didacticii și ale managementului clasei adaptate disciplinelor tehnice; A40 — gestionarea clasei prin strategii de motivare, rezolvare a conflictelor și diferențiere a instruirii; R24 — transferul responsabil de cunoștințe tehnice adaptat profilului interlocutorilor; R40 — gestionarea autonomă a laboratorului de informatică, cu responsabilitate pentru securitatea digitală și prevenția comportamentelor inadecvate online. Conținuturile răspund cerințelor ARACIS pentru pregătirea psihopedagogică și standardelor MEN pentru cariera didactică în învățământul preuniversitar.

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 managementului clasei și a stilurilor de conducere; cunoașterea strategiilor de motivare, rezolvare a conflictelor și diferențiere a instruirii; identificarea elementelor de securitate digitală și de etică în mediul școlar online; capacitatea de a aplica un model teoretic	Colocviu scris: subiecte teoretice și un studiu de caz cu propunere de intervenție	50%

	la o situație concretă din laboratorul de informatică.		
8.5 Seminar/laborator	Calitatea contribuției la jocurile de rol și studiile de caz; coerența produselor elaborate la seminar (regulament de laborator, plan de intervenție anti-bullying, grilă de evaluare formativă); portofoliul cu lucrările de la cele 7 seminarii.	Evaluare continuă a participării și a portofoliului cu produsele realizate la seminar	50%

8.6 Standard minim de performanță:

Elaborarea unui plan concret de gestionare a unei situații specifice de management al clasei de informatică (la alegere: prevenția distragerii la laborator, intervenția într-un caz de cyberbullying, sau diferențierea instruirii într-o clasă eterogenă), cu identificarea principiului teoretic aplicabil și formularea a cel puțin trei pași concreți de intervenție. Un plan fără ancorare teoretică sau fără pași concreți nu îndeplinește standardul minim. Nota minimă de promovare: 5.

Data completării	Semnătura titularului de curs Lect. Dr. Popa Doinița	Semnătura titularului de seminar / laborator Lect. Dr. Popa Doinița
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: 2029-2030

1. Date despre disciplină

1.1 Denumirea disciplinei	PRACTICĂ PEDAGOGICĂ ÎN ÎNV. PREUNIVERSITAR OBLIGATORIU (2) – TIDFAC425						
1.2 Titularul activităților de curs							
1.3 Titularul activităților de seminar							
1.4 Titularul activităților de laborator	Conf. Dr. Beteringhe Adrian						
1.5 Anul de studiu	IV	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		2.3 seminar/laborator	0/3
2.4 Total ore din planul de învățământ	42	din care: 2.5 curs		2.6 seminar/laborator	0/42
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	Pedagogie I, Pedagogie II, Psihologia educației, Didactica specializării (informatică), Practică pedagogică în învățământ preuniversitar obligatoriu (1), Managementul clasei de elevi.
3.2 de competențe	Cunoștințe de psihopedagogie (teorii ale învățării, metode didactice); capacitatea de a proiecta o lecție; familiaritate cu programa școlară de informatică/TIC; experiența observării lecțiilor din practica pedagogică (1).

4. Condiții (acolo unde este cazul)

4.1 de desfășurare a cursului	
4.2 de desfășurare a seminarului/laboratorului	Unitate de aplicație din învățământul preuniversitar (școală gimnazială sau liceu) cu profil real sau cu disciplină de informatică/TIC; mentor — profesor titular de specialitate; laborator de informatică funcțional; acces la programa școlară, manuale și auxiliare didactice.

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

5.1 Cunoștințe	C35. Posedă cunoștințe din domeniul psihologiei educației și al pedagogiei — teorii ale învățării, metode didactice, proiectare curriculară — necesare activității de instruire și transfer de cunoștințe tehnice specializate. C36. Cunoaște principiile didacticii specializării și ale managementului clasei, inclusiv strategii de evaluare formativă și summativă, adaptate predării disciplinelor tehnice din învățământul preuniversitar și universitar.
5.2 Aptitudini	A39. Aplică tehnici pedagogice și didactice adecvate în activitățile de predare, îndrumare și mentoring tehnic, proiectând activități de învățare diferențiată și evaluând progresul cursanților prin metode formative și sumative. A40. Gestionează clasa de elevi în contexte de educație tehnică, aplicând strategii de motivare, rezolvare a conflictelor și diferențiere a instruirii, asigurând un climat favorabil învățării disciplinelor informatice.
5.3 Responsabilitate și autonomie	R39. Acționează cu responsabilitate pedagogică și etică în toate activitățile didactice, respectând principiile echității, incluziunii și respectului față de diversitatea cursanților, și sesizând prompt situațiile care necesită intervenție specializată. R40. Gestionează autonom clasa de elevi în activitățile practice de informatică, asumând responsabilitatea pentru securitatea digitală a mediului de lucru al elevilor, respectarea regulamentelor școlare și prevenirea comportamentelor inadecvate în mediul online.

6. Conținuturi

6.1 Laborator	Nr. ore	Metode de predare	Observații
Lucrarea L1: Cunoașterea unității de aplicație și a clasei. Prezentarea școlii, a profesorului mentor, a regulamentului intern și a programei de informatică/TIC. Studierea documentelor didactice ale mentorului: planificarea anuală, planificările pe unități de învățare, manualul utilizat. Observarea componentei clasei și a profilului elevilor.	6	Observare, proiectare didactică, predare efectivă	
Lucrarea L2: Observarea lecțiilor mentorului și analiza didactică. Asistarea la cel puțin trei lecții ale profesorului mentor cu fișă de observație structurată: obiective, metode, mijloace, gestionarea clasei, evaluare. Discuții post-lecție cu mentorul. Redactarea fișelor de observație în portofoliu.	6	Observare, proiectare didactică, predare efectivă	
Lucrarea L3: Proiectarea primei lecții susținute. Elaborarea proiectului didactic complet (obiective operaționale, scenariu, fișe de lucru, instrumente de evaluare) pentru o lecție de informatică/TIC din programa curentă. Discutarea proiectului cu mentorul și revizuire pe baza feedback-ului. Pregătirea materialelor și a mediului din laborator.	6	Observare, proiectare didactică, predare efectivă	
Lucrarea L4: Susținerea lecției 1 și debriefing. Predarea efectivă a lecției proiectate la L3 în fața clasei, sub supravegherea mentorului. Înregistrarea observațiilor în jurnalul de practică. Analiza post-lecție împreună cu mentorul: puncte tari, dificultăți întâmpinate, ajustări pentru viitor. Redactarea raportului de autoevaluare.	6	Observare, proiectare didactică, predare efectivă	
Lucrarea L5: Proiectarea și susținerea lecției 2 cu diferențiere a instruirii. Elaborarea unui proiect didactic care include sarcini diferențiate pentru elevi cu niveluri diferite (începători-avansați la	6	Observare, proiectare didactică, predare efectivă	

programare). Susținerea lecției și aplicarea unei evaluări formative scurte. Analiza rezultatelor și a feedback-ului primit de la elevi.			
Lucrarea L6: Proiectarea și susținerea lecției 3 cu evaluare summativă. Elaborarea unei lecții de fixare/evaluare cu instrumente complete: fișă de evaluare, barem de corectare, grilă de notare. Susținerea lecției și administrarea evaluării. Corectarea lucrărilor și analiza rezultatelor pe clase de erori. Discutarea rezultatelor cu mentorul.	6	Observare, proiectare didactică, predare efectivă	
Lucrarea L7: Activități extracurriculare, finalizarea portofoliului și colocviul. Participarea la o activitate extracurriculară (cerc de informatică, concurs, ședință cu părinții). Finalizarea portofoliului de practică: fișe de observație, proiecte didactice, autoevaluări, fișa de evaluare a mentorului. Susținerea colocviului — prezentarea unei lecții reprezentative și reflecție asupra practicii.	6	Observare, proiectare didactică, predare efectivă	
<p>Bibliografie obligatorie laborator</p> <p>1. Iucu, R.B., Instruirea școlară. Perspective teoretice și aplicative, Polirom, 2008.</p> <p>2. Pânișoară, I.O., Profesorul de succes. 59 de principii de pedagogie practică, ed. a 2-a, Polirom, 2017.</p> <p>3. Manolescu, M., Teoria și metodologia evaluării, Editura Universitară, 2010.</p> <p>Bibliografie complementara laborator</p> <p>4. Cucuș, C., Pedagogie, ed. a 3-a, Polirom, 2014.</p> <p>5. Cerghit, I., Metode de învățământ, ed. a 4-a, Polirom, 2018.</p> <p>6. Iucu, R.B., Managementul clasei de elevi, Polirom, 2006.</p> <p>Resurse online gratuite recomandate</p> <p>7. Programa școlară de Informatică și TIC: https://www.edu.ro/programe-scolare</p> <p>8. Manuale digitale aprobate MEN: https://manuale.edu.ro/</p> <p>9. ISE — resurse pentru cariera didactică: https://ise.ro/</p> <p>10. Edupedu — știri din educație: https://www.edupedu.ro/</p>			

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

Practica pedagogică (2) consolidează competențele formate la practica (1) și asigură trecerea de la observare la predare efectivă: C35 și C36 sunt aplicate în proiectarea celor trei lecții susținute în clasă; A39 — proiectarea învățării diferențiate (L5) și a evaluării sumative (L6); A40 — gestionarea clasei pe parcursul lecțiilor predate; R39 — etica și echitatea în relația cu elevii reali; R40 — responsabilitatea pentru securitatea digitală în laboratorul de informatică. Disciplina răspunde cerințelor MEN pentru obținerea certificatului DPPD nivel I și standardelor ARACIS pentru pregătirea inițială a profesorilor de informatică/TIC.

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	Calitatea proiectelor didactice elaborate; eficiența predării lecțiilor susținute (claritatea explicațiilor, gestionarea clasei, gradul de implicare al elevilor, atingerea obiectivelor); coerența și completitudinea portofoliului de	Evaluare continuă pe parcursul celor 3 lecții susținute (observare cu fișă structurată); evaluare finală a portofoliului de practică și a colocviului de la L7; fișa de evaluare a profesorului mentor	100%

	practică; capacitatea de autoevaluare și reflecție. Evaluarea finală a mentorului are pondere importantă.		
8.6 Standard minim de performanță:			
Susținerea efectivă a celor trei lecții prevăzute (L4, L5, L6), cu proiecte didactice complete și obiective operaționale corect formulate, conform taxonomiei Bloom; predarea unei lecții cu sarcini diferențiate pentru cel puțin două niveluri de elevi; aplicarea unei evaluări summativă cu barem transparent; depunerea unui portofoliu complet (proiecte didactice, fișe de observație, autoevaluări, fișa de evaluare a mentorului). Absența unei lecții susținute sau a portofoliului complet nu îndeplinește standardul minim. Nota minimă de promovare: 5.			

Data completării	Semnătura titularului de curs	Semnătura titularului de seminar / laborator Conf. Dr. Beteringhe Adrian
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		