FIŞA
DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Braşov
1.2 Facultatea	Matematică şi Informatică
1.3 Departamentul	Matematică Informatică
1.4 Domeniul de studii de Licentă [1)]	Informatică
1.5 Ciclul de studii[2)]	Licentă
1.6 Programul de studii/ Calificarea	Informatică aplicată / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Verificarea și validarea sistemelor software							
2.2 Titularul activităților de curs				Conf. dr. Cârstea Claudia				
2.3 Titularul activităților de seminar/ laborator/proiect				Conf. dr. Cârstea Claudia				
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[3)]	DI

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 seminar/ laborator/ proiect	**0/2/0**
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar/ laborator/proiect	0/28/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie şi notițe	35
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	20
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	25
Tutoriat	15
Examinări	4
Alte activități.....	

3.7 Total ore de studiu individual	94
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Nu este cazul
4.2 de competențe	• Cunoașterea tehnicilor de organizare date • Deprinderi de utilizare calculator nivel avansat

5. Condiții (acolo unde este cazul)

5.1 de desfăşurare a cursului	• videoproiector
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	• laborator de informatică

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențeprofesionale	CP2. Gestionarea ciclului de viață al aplicațiilor informatice **Cunoștințe** R.Î.2.1. Absolventul poate să parcurgă ciclul de viață al unui sistem informatic, începând cu ancheta preliminară și terminând cu exploatarea și menținerea sistemului. R.Î.2.2. Absolventul poate să identifice și să evalueze riscurile asociate cu fiecare fază a ciclului de viață. **Aptitudini** R.Î.2.3. Absolventul poate să proiecteze soluții software eficiente și scalabile pentru fiecare fază a ciclului de viață. R.Î.2.4. Absolventul poate să implementeze și să testeze componente software conform specificațiilor funcționale și non-funcționale. R.Î.2.5. Absolventul poate să monitorizeze și să optimizeze performanța sistemului în faza de exploatare. R.Î.2.6. Absolventul poate să aplice măsuri de securitate pentru protejarea datelor pe tot parcursul ciclului de viață. R.Î.2.7. Absolventul poate realiza documentație tehnică și manuale de utilizare pentru aplicațiile software, facilitând astfel utilizarea și întreținerea acestora de către specialiști din diverse domenii. **Responsabilitate și autonomie** R.Î.2.8. Absolventul poate să lucreze în echipă pentru a parcurge ciclul de viață al unui sistem informatic. CP3. Utilizarea instrumentelor informatice în context interdisciplinar **Cunoștințe** R.Î.3.1. Absolventul poate utiliza instrumente informatice (folosind tehnologii specifice) pentru a colecta, analiza și interpreta date din diferite domenii de activitate. R.Î.3.2. Absolventul poate evalua și selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare, având abilitatea de a analiza nevoile specifice ale unui proiect și de a alege soluții tehnologice eficiente. **Aptitudini** R.Î.3.3. Absolventul poate implementa metode de testare și evaluare a software-ului, inclusiv testarea interactivă și feedback-ul utilizatorilor din diverse discipline, pentru a asigura calitatea și funcționalitatea aplicațiilor create. **Responsabilitate și autonomie** R.Î.3.5. Absolventul poate să comunice și să colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice, dezvoltând competențe de lucru în echipă și abilități de gestionare a proiectelor interdisciplinare.
Competențetransversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională CT2 Desfăşurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoştințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională

7. Obiectivele disciplinei (reieşind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	• Familiarizarea cu instrumentele de VVSS, informații, cunoștințe, societateinformațională, decizii, tipuri de decizii, luarea deciziilor • Conținutul teoretic al cursului va fi consolidat prin aplicații practice în cadrul laboratorului • Verificarea logică a sistemelor şi subsistemelor informatice

7.2 Obiectivele specifice	• Formularea și punerea în aplicare a sistemelor suport de decizie, în general, șiîn afaceri în special – verificare si validare sisteme

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
Validare vs Verificare. Sistem Informatic. Sistem Informațional.	prelegerea, explicația, demonstrația, problematizarea,exercițiul	2	
Definirea şi caracterizarea sistemelor. Elemente componente ale unui sistem informatic.		2	
Ciclul de viață al unui sistem informatic		2	
Metodologii de analiză şi proiectare a unui sistem informatic		2	
Conținutul metodologiilor de realizare a unui SI. Verificare activități.		2	
Strategii de abordare a etapelor de analiză şi proiectare a unui SI		4	
Instrumente pentru verificare și validare software		6	
VVSS – Instrumente open source		4	
Tehnici de testare		2	
Tehnici de validare		2	

Bibliografie

- Claudia Cârstea – suport de curs - 2022
- Claudia Cârstea – Sisteme informatice pentru asistarea deciziei, ed. Omnia UNI S.A.S.T, Brasov, 2008, ISBN 978-973-134-045-6, (444 pagini A5)
- Claudia Cârstea – Sisteme informatice de gestiune, ed. Omnia UNI S.A.S.T, Brasov, 2007, ISBN 978-973-134-034-0,(342 pagini A5)
- Claudia Cârstea – Sisteme informa ionale şi aplica ii informatice în administrarea afacerilor, ed. Omnia UNIS.A.S.T, Brasov, 2007, ISBN 978-973-134-005-0, (424 pagini A5)
- Claudia Cârstea – Proiectarea bazelor de date. Mediul de programare Microsoft Access, ed. Omnia UNI S.A.S.T,Brasov, 2007, ISBN 978-973-134-000-5, (324 pagini A5)
- Gluchowski et al., Management Support Systeme und Business Intelligence,2008
- Turban E, Aronson J.E., Ting-Peng Liang Decision Support Systems and Intelligent Systems, Prentice Hall, 2005, 7thEdition.
- F Brustein, C.W.Hosaple, Handbook on Decision Support Systems 1, Basic Themes Springer, 2008.
- Gupta J.N.D et al., Intelligent Decision-making support systems, Foundations, Applications and Chalanges,Springer, 2006.
- Filip F., Decision support systems supported by computers, Editura Tehnică, 2005.
- Filip F., Decision Support Systems, Editura Tehnică, 2007.
- Brândaş C., Decisions support systems for management, Brumar, 2007.
- Rusu L., Arba R., Bresfelean P., MuresanL. Veres O., Integrated information systems and ERPs, Risoprint,

2005

8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore	Observații
Verificare și validare sistem informaticpentru evidența studentâților pe specializări.	Exercițiul, explicația, problematizarea, demonstrația	4	
Verificare și validare pentru sistemul de contractare, livrare, încasare produse.		4	
Verificare și validare sistem informatic pentru evidența abonaților dintr-o bibliotecă.		4	
Verificare și validare sistem informatic pentru managementul resurselor umane.		4	
Verificare și validare sistem informatic pentru evidența rezervărilor la o unitate hotelieră.		4	
Prezentare şi sustinere proiecte în echipe		8	

Bibliografie

- Claudia Cârstea – suport de curs - 2022
- Claudia Cârstea – Sisteme informatice pentru asistarea deciziei, ed. Omnia UNI S.A.S.T, Brasov, 2008, ISBN 978-973-134-045-6, (444 pagini A5)
- Claudia Cârstea – Sisteme informatice de gestiune, ed. Omnia UNI S.A.S.T, Brasov, 2007, ISBN 978-973-134-034-0,(342 pagini A5)
- Claudia Cârstea – Sisteme informa ionale şi aplica ii informatice în administrarea afacerilor, ed. Omnia UNIS.A.S.T, Brasov, 2007, ISBN 978-973-134-005-0, (424 pagini A5)
- Claudia Cârstea – Proiectarea bazelor de date. Mediul de programare Microsoft Access, ed. Omnia UNI S.A.S.T,Brasov, 2007, ISBN 978-973-134-000-5, (324 pagini A5)
- Gluchowski et al., Management Support Systeme und Business Intelligence,2008
- Turban E, Aronson J.E., Ting-Peng Liang Decision Support Systems and Intelligent Systems, Prentice Hall, 2005, 7thEdition.
- F Brustein, C.W.Hosaple, Handbook on Decision Support Systems 1, Basic Themes Springer, 2008.
- Gupta J.N.D et al., Intelligent Decision-making support systems, Foundations, Applications and Chalanges,Springer, 2006.
- Filip F., Decision support systems supported by computers, Editura Tehnică, 2005.
- Filip F., Decision Support Systems, Editura Tehnică, 2007.
- Brândaş C., Decisions support systems for management, Brumar, 2007.
- Rusu L., Arba R., Bresfelean P., MuresanL. Veres O., Integrated information systems and ERPs, Risoprint, 2005

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, ale asociaților profesionale şi ale angajatorilor reprezentativi din domeniul aferent programului

Continuturile sunt în concordantă cu problematica monografiilor apărute în ultimii ani în literatura de specialitate.

10. Evaluare

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Cunoașterea noțiunilor privind ciclul de viată al unui sistem si modalitățile de abordare Proiectarea unui sistem sau subsistem informatic si declararea interogărilor	Test grilă	50%
10.5 Seminar/ laborator/ proiect	Realizarea unei biblioteci cu aplicații rezolvate la laborator	Calificativ admis/respins	30%
	Rezolvare de interogări in SQL	Calificativ admis/respins	20%
10.6 Standard minim de performanță			
• capacitatea de a utiliza conceptele prezentate in diverse etape de analiza si proiectare a sistemelor informatice,precum si declararea interogărilor in SQL. • calitatea prezentării referatului si argumentarea temei acestuia.			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. dr. Ion Gabriel Stan	Director de Departament Conf. dr. Nicușor Minculete
Titular curs Conf. dr. Cârstea Claudia	Titular laborator Conf. dr. Cârstea Claudia

Notă:

1) Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

2) Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

3) Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

4) Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii de licență[1)]	Informatică
1.5 Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	INFORMATICĂ APLICATĂ/Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Tehnologii WEB							
2.2 Titularul activităților de curs				Conf. dr. Dumitrescu Silviu				
2.3 Titularul activităților de seminar/ laborator/ proiect				Drd. Rîtan Lucian				
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[3)]	DI

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

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar/ laborator/ proiect	0/28/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie şi notițe	35
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	20
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	20
Tutoriat	10
Examinări	9
Alte activități.....	0
3.7 Total ore de studiu individual	94
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Programare orientată pe obiect
4.2 de competențe	Competențe informatice de nivel preuniversitar

5. Condiții (acolo unde este cazul)

5.1 de desfăşurare a cursului	Sala de curs cu videoproiector
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	Java 17, Spring 3+

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt **Cunoștințe** R.Î.1.1. Absolventul poate să scrie cod clar și bine structurat în limbaje de programare specifice. R.Î.1.2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale. **Aptitudini** R.Î.1.3. Absolventul poate să utilizeze biblioteci și framework-uri pentru îmbunătățirea performanțelor și funcționalitatea aplicațiilor software. R.Î.1.4. Absolventul poate să dezvolte aplicații complexe cu cerințele propuse de către utilizatori. **Responsabilitate și autonomie** R.Î.1.5. Absolventul este capabil să organizeze, planifice și supravegheze dezvoltarea aplicațiilor și a cadrelor necesare pentru a crea un produs software complet, de la etapele inițiale de planificare până la testarea finală. CP3. Utilizarea instrumentelor informatice în context interdisciplinar **Cunoștințe** R.Î.3.1. Absolventul poate utiliza instrumente informatice (folosind tehnologii specifice) pentru a colecta, analiza și interpreta date din diferite domenii de activitate. R.Î.3.2. Absolventul poate evalua și selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare, având abilitatea de a analiza nevoile specifice ale unui proiect și de a alege soluții tehnologice eficiente. **Aptitudini** R.Î.3.3. Absolventul poate implementa metode de testare și evaluare a software-ului, inclusiv testarea interactivă și feedback-ul utilizatorilor din diverse discipline, pentru a asigura calitatea și funcționalitatea aplicațiilor create. **Responsabilitate și autonomie** R.Î.3.5. Absolventul poate să comunice și să colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice, dezvoltând competențe de lucru în echipă și abilități de gestionare a proiectelor interdisciplinare. CP7. Analiza, proiectarea, dezvoltarea, administrarea și mentenanța sistemelor software pentru WEB şi Internet **Cunoștințe** R.Î.7.1. Absolventul poate să identifice și să evalueze cerințelor sistemului software. R.Î.7.2. Absolventul poate să analizeze și să proiecteze arhitectura sistemului software. **Aptitudini** R.Î.7.3. Absolventul poate să identifice și să rezolve problemele de performanță și scalabilitate. R.Î.7.4. Absolventul poate să proiecteze interfețe WEB pentru utilizator, integrând baze de date și algoritmi eficienți pentru procesarea datelor. R.Î.7.5. Absolventul poate să administreze infrastructura hardware și software a sistemelor WEB și să monitorizeze performanța și securitatea sistemelor WEB. R.Î.7.6. Absolventul poate să remedieze erorilor și vulnerabilitățile de securitate. R.Î.7.7. Absolventul poate să asigure compatibilitatea cu noile versiuni de tehnologii WEB. **Responsabilitate și autonomie** R.Î.7.8. Studentul/absolventul produce software și îl adaptează continuu la noile tehnologii și cerințe de piață.

Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoştințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională

7. Obiectivele disciplinei (reieşind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	Dobândirea de cunoștințe de dezvoltare ale aplicațiilor web, având ca suport limbajul Java.
7.2 Obiectivele specifice	Formarea unor deprinderi relativ la securizarea aplicațiilor, arhitectura web, bune practici

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
Programarea într-o rețea multiclient folosind socket si serversocket	Expunere	6	
Programarea într-un mediu distribuit web folosind servlet		4	
Arhitecturi web folosind Spring/Sprint Boot		12	
Persistența în Spring folosind Hibernate		6	
Bibliografie: 1. DANCIU D., DUMITRESCU S., Dezvoltarea aplicațiilor Java, Ed. Universității Transilvania, 2008 2. https://spring.io/guides 3. https://www.springboottutorial.com/			
8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore	Observații
Aplicații folosind java.net	Lucrări practice	4	
Aplicații folosind servlet		4	
Aplicații Spring boot		4	
Proiect ce prezintă un flux complet web		16	
Bibliografie: 1. DANCIU D., DUMITRESCU S., Dezvoltarea aplicațiilor Java, Ed. Universității Transilvania, 2008 2. https://spring.io/guides 3. https://www.springboottutorial.com/			

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, ale asociaților profesionale şi ale angajatorilor reprezentativi din domeniul aferent programului

Coroborarea se aplică în acordurile de parteneriat și în Convențiile de practică încheiate cu partenerii din mediul socio-economic al Facultății/Universității

10. Evaluare

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Atingerea obiectivelor educaționale ale disciplinei	Examinare practica si test grila	85%
10.5 Seminar/ laborator/ proiect		Teme laborator	15%
10.6 Standard minim de performanță			
Curs: cunoașterea și înțelegerea framework web Laborator: implementarea de aplicații web, cu cerințele de calitate date de arhitecturile enterprise.			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. dr. Ion Gabriel Stan	Director de departament Conf. dr. Nicușor Minculete
Titular de curs Conf. dr. Dumitrescu Silviu	Titular de seminar/laborator/proiect Drd. Rîtan Lucian

Notă:

1) Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

2) Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

3) Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

4) Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Braşov
1.2 Facultatea	Matematică şi Informatică
1.3 Departamentul	Matematică şi Informatică
1.4 Domeniul de studii de licență	Informatică
1.5 Ciclul de studii	Licență
1.6 Programul de studii/ Calificarea	Informatică aplicată / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	**Dezvoltarea Aplicațiilor Mobile**							
2.2 Titularul activitătilor de curs				Conf. Dr. Răzvan Bocu				
2.3 Titularul activităților de laborator				Conf. Dr. Răzvan Bocu				
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei	Continut	DS
							Obligativitate[3)]	DI

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 seminar/ laborator/ proiect	**0/2/0**
3.4 Total ore din planul de învătământ	56	din care: 3.5 curs	28	3.6 seminar/ laborator/ proiect	0/28/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie şi notițe	33
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	10
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	40
Tutoriat	7
Examinări	4
Alte activități.....	0

3.7 Total ore studiu individual	94
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Preconditii

4.1 de curriculum	Programare orientată pe obiecte, instrumente de programare și concepte specifice programării dispozitivelor mobile, baze de date relaționale și nonrelaționale
4.2 de competențe	Descrierea adecvată a paradigmelor de programare şi a mecanismelor de limbaj specifice, precum şi identificarea diferenței dintre aspectele de ordin semantic şi sintactic; Dezvoltarea de unități de program şi elaborarea documentațiilor aferente; Identificarea modelelor si metodelor adecvate pentru rezolvarea unor probleme reale.

5. Conditii

5.1 de desfăşurare a cursului	Proiector, laptop, tabla
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	IDE Android Studio, Java SDK, Android SDK, Android Development Tools

6. Competente specifice acumulate

Competente profesionale	CP1 Programarea în limbaje de nivel înalt R.Î. 1.1. Absolventul poate să scrie cod clar si bine structurat in limbaje de programare specifice. R.Î. 1.2. Absolventul poate să utilizeze biblioteci și framework-uri pentru îmbunătățirea performantelor si funcționalitatea aplicațiilor software. R.Î. 1.3. Absolventul poate să dezvolte aplicatii complexe cu cerintele propuse de catre utilizatori R.Î. 1.4. Absolventul poate să aplice metodele şi principiile folosind tehnologii specifice CP2 Parcurgerea ciclului de viață pentru aplicațiile informatice R.Î. 2.1. Absolventul poate să parcurge ciclul de viață al unui sistem informatic, începând cu ancheta preliminară și terminând cu exploatarea și menținerea sistemului. R.Î. 2.2. Absolventul poate să identifice și să evalueze riscurile asociate cu fiecare fază a ciclului de viață. R.Î. 2.3. Absolventul poate sa lucreze în echipă pentru a a parcurge ciclul de viață al unui sistem informatic CP3 Utilizarea instrumentelor informatice în context interdisciplinar R.Î. 3.1. Absolventul poate utiliza instrumente informatice (folosind tehnologii specifice) pentru a colecta, analiza și interpreta date din diferite domenii de activitate R.Î. 3.2. Absolventul poate evalua și a selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare R.Î. 3.3. Absolventul poate să comunice și sa colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice. CP4 Utilizarea bazelor teoretice ale informaticii și a modelelor formale R.Î. 4.1. Absolventul poate să utilizeze modele formale și teorii matematice pentru a rezolva probleme din diverse domenii ale informaticii R.Î. 4.2. Absolventul poate sa utilizeze concepte și tehnici matematice pentru a analiza și proiecta algoritmi și structuri de date folosind tehnologii specifice. R.Î. 4.3. Absolventul poate să utilizeze modele formale pentru a proiecta sisteme informatice sigure și fiabile. R.Î. 4.4. Absolventul poate să utilizeze modele formale pentru a verifica corectitudinea programelor. CP5 Proiectarea, dezvoltarea şi gestiunea bazelor de date R.Î. 5.1. Absolventul poate să proiecteze scheme de baze de date eficiente și bine structurate. R.Î. 5.2. Absolventul poate să a utiliza limbaje de interogare pentru a accesa și manipula datele din bazele de date R.Î. 5.3. Absolventul poate să implementeze și sa gestioneze sisteme de gestiune a bazelor de date (SGBD). R.Î. 5.4. Absolventul poate să asigure integritatea și securitatea datelor din bazele de date. R.Î. 5.5. Absolventul poate să a optimizeze performanța bazelor de date prin indexare și alte tehnici relevante. CP6 Proiectarea, dezvoltarea, administrarea si asigurarea securității rețelelor de calculatoare R.Î. 6.1. Absolventul poate să stabilească configurarea și administrarea rețelelor de calculatoare R.Î. 6.2. Absolventul poate să stabilească configurarea și administrarea principalele protocoale de rutare R.Î. 6.3. Absolventul poate să stabileasca metodele de structurare și administrare a dispozitivelor de rețe R.Î. 6.4. Absolventul poate să verifice aplicatiile de administrarea retelelor de calculatoare folosind CP7 Analiza, proiectarea, dezvoltarea, administrarea si mentenanța sistemelor software pentru WEB şi Internet. R.Î. 7.1 Absolventul poate să identifice și să evalueze cerințelor sistemului software. R.Î. 7.2 Absolventul poate să analizeze și si să proiecteze arhitectura sistemului software. R.Î. 7.3 Absolventul poate să identifice și să rezolve problemele de performanță și scalabilitate R.Î. 7.4 Absolventul poate pentru aplicații WEB sa proiecteze interfața pentru utilizator, pentru bazele de date și pentru algoritmilor eficienți pentru procesarea datelor. R.Î. 7.5 Absolventul poate sa administreze infrastructuria hardware și software a sistemelor web si sa monitorizeze performanța și securitatea sistemelor web. R.Î. 7.6 Absolventul poate să remedieze erorilor și vulnerabilitățile de securitate.

Competente transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. R.Î.1.3. Absolventul are capacitatea de a acționa în mod responsabil și etic în activitățile didactice și de cercetare, demonstrând integritate și respect față de colegi, studenți și comunitatea academică. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite. CT2 Desfăşurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. R.Î.2.3. Absolventul poate aplica tehnici de comunicare empatică, ascultând activ și răspunzând adecvat nevoilor și preocupărilor colegilor, contribuind astfel la crearea unui mediu de lucru armonios. R.Î.2.4. Absolventul poate adapta stilul de comunicare în funcție de contextul cultural și diversitatea grupului, asigurându-se că mesajul este transmis eficient și este bine înțeles de toți participanții.

7. Obiectivele disciplinei

7.1 Obiectivul general al disciplinei	Programarea orientată pe obiecte în contextul platformei Android Dezvoltarea de interfețe grafice pentru dispozitivele mobile Dezvoltarea aplicațiilor mobile avansate
7.2 Obiectivele specifice	Prezentarea platformei Android Proiectarea şi implementarea aplicațiilor Android în limbajul Java Proiectarea şi implementarea interfețelor Android Prezentarea conceptelor avansate relevante ale limbajului Java Proiectarea şi implementarea bazelor de date pentru aplicațiile Android

8. Continuturi

8.1 Curs	Metode de predare	Număr de ore	Observatii
Android SDK şi Android Studio	Expunere, informare	2	
Concepte avansate Java	Expunere, explicație, exemplificare	2	
Proiectarea şi implementarea interfețelor Android	Expunere, explicație, exemplificare	2	
Fragment şi Intent	Expunere, explicație, exemplificare	2	
Asigurarea compatibilității cu versiunile precedente	Expunere, explicație, exemplificare	2	
Baze de date în Android	Expunere, explicație, exemplificare	2	

Senzori şi acțiuni / gesturi	Expunere, explicație, exemplificare	2	
Multimedia, localizare, comunicare, NFC	Expunere, explicație, exemplificare	4	
Fire de execuție	Expunere, explicație, exemplificare	2	
Android în contextul aplicațiilor distribuite şi de tip Cloud	Expunere, explicație, exemplificare	4	
Servicii	Expunere, explicație, exemplificare	4	
Bibliografie [1] Horton, J. Android Programming for Beginners: Build in-depth, full-featured Android apps starting from zero programming experience. Packt Publishing Ltd., 2021. [2] Boudjnah, E. Clean Architecture for Android: Implement Expert-led Design Patterns to Build Scalable, Maintainable, and Testable Android Apps (English Edition). BPB Publications, 2022.			
8.2 Seminar/ laborator/ proiect	Metode de predare-învă are	Număr de ore	Observatii
Configurarea mediului de lucru cu ajutorul Android Studio	Exemplificare, proiecte, teme de laborator	2	-
Primele aplicații	Exemplificare, proiecte, teme de laborator	2	
Construirea unui view	Exemplificare, proiecte, teme de laborator	4	
Grafică 2D	Exemplificare, proiecte, teme de laborator	2	-
Grafică 3D	Exemplificare, proiecte, teme de laborator	4	-
Utilizarea bazelor de date în Android	Exemplificare, proiecte, teme de laborator	4	-
Multimedia	Exemplificare, proiecte, teme de laborator	4	-
Localizare	Exemplificare, proiecte, teme de laborator	2	-
Senzori şi comunicare	Exemplificare, proiecte, teme de laborator	2	-
Rularea codului nativ cu Android NDK	Exemplificare, proiecte, teme de laborator	2	-
Bibliografie [1] Boudjnah, E. Clean Architecture for Android: Implement Expert-led Design Patterns to Build Scalable, Maintainable, and Testable Android Apps (English Edition). BPB Publications, 2022. [2] Smyth, N. Android Studio Hedgehog Essentials-Kotlin Edition: Developing Android Apps Using Android Studio 2023.1. 1 and Kotlin. Payload Publishing, 2023.			

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, asociațiilor profesionale şi angajatori reprezentativi din domeniul aferent programului

Coroborarea se aplică in *Acordurile de parteneriate* și in *Conventiile de Practică* încheiate cu partenerii din mediul socio-economic al Facultății/Universității. Fișa disciplinei a fost elaborată după consultări cu partenerii din mediul academic și economic ai Facultății.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Cunoștinte teoretice specifice privind limbajul Java si platforma de operare Android	Evaluare finală prin susținerea unei prezentări ştiinţifice al cărei conținut este ilustrat prin intermediul unui sistem software Android	50%
10.5 Seminar/ laborator/ proiect	Acumularea competențelor vizate prin conținutul disciplinei	Evaluare continuă/formativă prin proiecte și activități participative.	50%
10.6 Standard minim de performantă Laborator: Implementarea minimală a aplicațiilor Android care presupun utilizarea activităților, fragmentelor, intențiilor, precum şi stocarea persistentă a datelor. Curs: Prezentarea elocventă a temei alocate, precum şi ilustrarea acesteia prin intermediul unei aplicații Android adecvate.			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. dr. Ion Gabriel Stan	Director de departament Conf. dr. Nicușor Minculete
Titular de curs Conf. dr. Răzvan Bocu	Titular de laborator Conf. dr. Răzvan Bocu

Notă:

[1)] Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

[2)] Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

[3)] Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

[4)] Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

[5)] Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii de licență[1)]	Informatică
1.5 Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	INFORMATICĂ APLICATĂ / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Practică de specialitate							
2.2 Titularul activităților de curs				Lect.dr. Nicoleta Enache-David				
2.3 Titularul activităților de seminar/ laborator/proiect				Lect.dr. Nicoleta Enache-David				
2.4 Anul de studiu	3	2.5 Semestrul	1	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[4)]	DI

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

3.1 Număr de ore pe săptămână	8	din care: 3.2 curs	-	3.3 seminar/ laborator/ proiect	0/0/8
3.4 Total ore din planul de învățământ	112	din care: 3.5 curs	-	3.6 seminar/ laborator/ proiect	0/0/112

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie și notițe	4
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	10
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri	20
Tutoriat	2
Examinări	2
Alte activități.....	0

3.7 Total ore de activitate a studentului	38
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	•
4.2 de competențe	•

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	• Nu este cazul
5.2 de desfășurare a seminarului/ laboratorului/ proiectului	• Studentul va fi coordonat de catre un mentor de practică

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențe profesionale	CP2. Gestionarea ciclului de viață al aplicațiilor informatice **Cunoștințe** R.Î.2.1. Absolventul poate să parcurgă ciclul de viață al unui sistem informatic, începând cu ancheta preliminară și terminând cu exploatarea și menținerea sistemului. R.Î.2.2. Absolventul poate să identifice și să evalueze riscurile asociate cu fiecare fază a ciclului de viață. **Aptitudini** R.Î.2.3. Absolventul poate să proiecteze soluții software eficiente și scalabile pentru fiecare fază a ciclului de viață. R.Î.2.4. Absolventul poate să implementeze și să testeze componente software conform specificațiilor funcționale și non-funcționale. R.Î.2.5. Absolventul poate să monitorizeze și să optimizeze performanța sistemului în faza de exploatare. R.Î.2.6. Absolventul poate să aplice măsuri de securitate pentru protejarea datelor pe tot parcursul ciclului de viață. R.Î.2.7. Absolventul poate realiza documentație tehnică și manuale de utilizare pentru aplicațiile software, facilitând astfel utilizarea și întreținerea acestora de către specialiști din diverse domenii. **Responsabilitate și autonomie** R.Î.2.8. Absolventul poate să lucreze în echipă pentru a parcurge ciclul de viață al unui sistem informatic. CP3. Utilizarea instrumentelor informatice în context interdisciplinar **Cunoștințe** R.Î.3.1. Absolventul poate utiliza instrumente informatice (folosind tehnologii specifice) pentru a colecta, analiza și interpreta date din diferite domenii de activitate. R.Î.3.2. Absolventul poate evalua și selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare, având abilitatea de a analiza nevoile specifice ale unui proiect și de a alege soluții tehnologice eficiente. **Aptitudini** R.Î.3.3. Absolventul poate implementa metode de testare și evaluare a software-ului, inclusiv testarea interactivă și feedback-ul utilizatorilor din diverse discipline, pentru a asigura calitatea și funcționalitatea aplicațiilor create. R.Î.3.4. Studentul/absolventul dezvoltă interesul pentru literatura de specialitate națională și internațională, participarea la cursuri specializate, conferințe, simpozioane, etc. pentru dezvoltarea profesională şi personală continuă și adaptarea eficientă la noile descoperiri ştiin ifice. **Responsabilitate și autonomie** R.Î.3.5. Absolventul poate să comunice și să colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice, dezvoltând competențe de lucru în echipă și abilități de gestionare a proiectelor interdisciplinare.

Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. R.Î.1.3. Absolventul are capacitatea de a acționa în mod responsabil și etic în activitățile didactice și de cercetare, demonstrând integritate și respect față de colegi, studenți și comunitatea academică. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite. CT2 Desfăşurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. R.Î.2.3. Absolventul poate aplica tehnici de comunicare empatică, ascultând activ și răspunzând adecvat nevoilor și preocupărilor colegilor, contribuind astfel la crearea unui mediu de lucru armonios. R.Î.2.4. Absolventul poate adapta stilul de comunicare în funcție de contextul cultural și diversitatea grupului, asigurându-se că mesajul este transmis eficient și este bine înțeles de toți participanții.

7. Obiectivele disciplinei (reieșind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	• Utilizarea metodologiilor şi a mediilor de dezvoltare pentru realizarea aplicațiilor informatice • Realizarea unor proiecte informatice dedicate • Dezvoltarea capacității studenților de a oferi consultanță pentru proiecte
7.2 Obiectivele specifice	• Dezvoltarea capacității studenților de a lucra în echipă în cadrul unui proiect informatic

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
Nu este cazul			
8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore	Observații
Practică la firmă/instituție; practică într-un laborator de cercetare din cadrul universitatii; practică în cadrul unui proiect din cadrul facultății.	Invățare prin proiecte	112	
Bibliografie			

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunităților epistemice, ale asociaților profesionale și ale angajatorilor reprezentativi din domeniul aferent programului

Fișa disciplinei a fost elaborată după consultări cu partenerii din mediul academic și economic ai Facultății.

10. Evaluare

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs			
10.5 Seminar/ laborator/ proiect	Aplicatie practică	Verificare proiect	100%
10.6 Standard minim de performanță			
Prezentarea documentelor: convenție de practică, caiet de practică, certificat de practică. Prezentarea proiectului/modulului realizat în cadrul stagiului de practică.			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. dr. Gabriel Stan	Director de departament Conf. dr. Nicușor Minculete
Titular curs Lect. dr. Nicoleta Enache-David	Titular laborator / seminar /proiect Lect. dr. Nicoleta Enache-David

Notă:

1) Domeniul de studii – se alege una din variantele: Licență / Masterat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor / programelor de studii universitare în vigoare).

2) Ciclul de studii – se alege una din variantele: Licență / Masterat/ Doctorat.

3) Regimul disciplinei (conținut); se alege una din variantele: DF (disciplină fundamentală) / DD (disciplină din domeniu) / DS (disciplină de specialitate) / DC (disciplină complementară) - pentru nivelul de licență ; DAP (disciplină de aprofundare) / DSI (disciplină de sinteză) / DCA (disciplină de cunoaștere avansată) - pentru nivelul de masterat.

4) Regimul disciplinei (obligativitate) – se alege una din variantele: DI (disciplină impusă) / DO (disciplină opțională) / DFac (disciplină facultativă).

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice și studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ	Universitatea *Transilvania* din Braşov
1.2 Facultatea	Matematică şi Informatică
1.3 Departamentul	Matematică şi Informatică
1.4 Domeniul de licență	Informatică
1.5 Ciclul de studii[1)]	Licență
1.6 Programul de studii/ Calificarea	Informatică Aplicată / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei				Deep Learning				
2.2 Titularul activităților de curs					Dr. Musat Bogdan-Adrian			
2.3 Titularul activităților de seminar/ laborator/ proiect					Dr. Musat Bogdan-Adrian			
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[2)]	DS
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs		2	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs		28	3.6 seminar/ laborator/ proiect	0/28/0

	Distribuția fondului de timp	ore
	Studiul după manual, suport de curs, bibliografie şi notițe	24
	Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	24
	Pregătire seminarii/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	42
	Tutoriat	0
	Examinări	4
	Alte activități.....	0

3.7 Total ore studiu individual	94	
3.8 Total ore pe semestru	150	
3.9 Numărul de credite	5	

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Machine Learning
4.2 de competențe	

5. Condiții (acolo unde este cazul)

5.1 de desfăşurare a cursului	Proiector, laptop, tabla
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	Python, interfata de programare PyCharm

6. Competențe specifice acumulate

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt CP2. Gestionarea ciclului de viață al aplicațiilor informatice CP3. Utilizarea instrumentelor informatice în context interdisciplinar CP4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale
Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională. CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar și dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare și colaborare cu grupuri diverse CT3 Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare în limba română și într-o limbă de circulație internațională

7. Obiectivele disciplinei (reieşind din grila competențelor specifice acumulate)

7.1 Obiectivul general al disciplinei	Modele de Deep Learning Metode de învățare automata
7.2 Obiectivele specifice	Învățare supervizata, nesupervizata si prin întărire Citirea si scrierea unui articol științific

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore
Introducere in Deep Learning	Expunere Dialog Explicație Dezbatere Informare Exemple	2
Noțiuni preliminare de matematica		2
Rețele neuronale		2
Rețele neuronale convolutionale		2
Rețele neuronale recurente		2
Transformers		4
Unsupervised learning		4
Generative adversarial networks		4
Self-supervised and semi-supervised learning		2
Reinforcement learning		2
Paper reading and writing techniques		2
		Total 28 de ore

Bibliografie: https://www.deeplearningbook.org/ https://mml-book.github.io/ https://huggingface.co/		
8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore
Retele neuronale	Exerciții Dialog Lucru în echipă Problematizare Studiu individual	2
Retele neuronale convolutionale		2
Retele neuronale recurente		2
Transformers		4
Unsupervised learning		4
Generative adversarial networks		4
Self-supervised and semi-supervised learning		4
Reinforcement learning		2
Studiu de caz		2
		2
		Total 28 de ore
Bibliografie https://www.deeplearningbook.org/ https://mml-book.github.io/ https://huggingface.co/		

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, asociaților profesionale şi angajatori reprezentativi din domeniul aferent programului

Coroborarea se aplică în *Acordurile de parteneriat* şi în *Convențiile de Practică* încheiate cu partenerii din mediul socio-economic ai Facultății/Universității. Fișa disciplinei a fost elaborată după consultări cu partenerii din mediul academic și economic ai Facultății.

10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
10.4 Curs	Atingerea obiectivelor educaționale ale disciplinei	Examen final prin proba scrisă	50%

	Formarea competențelor vizate prin conținutul disciplinei		
10.5 Seminar/ laborator/ proiect		Evaluare continuă/formativă prin proiecte și activități participative.	50%

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Conf. Dr. Ion Gabriel Stan Decan	Conf. Dr. Nicușor Minculete Director de departament
Dr. Musat Bogdan-Adrian Titular de curs	Dr. Musat Bogdan-Adrian Titular de laborator

Notă:

[1)] Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

[2)] Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

[3)] Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

[4)] Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

[5)] Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii de licență[1)]	Informatică
1.5 Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	INFORMATICĂ APLICATĂ/ Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Procesarea imaginilor digitale							
2.2 Titularul activităților de curs				Lect. dr. Plajer Ioana Cristina				
2.3 Titularul activităților de laborator				Lect. dr. Plajer Ioana Cristina Drd. Cristian George Fieraru				
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6 Tipul de evaluare	C	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar/ laborator/ proiect	0/28/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie și notițe	33
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	10
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri	40
Tutorat	7
Examinări	4
Alte activități.....	0

3.7 Total ore studiu individual	94
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Algoritmi, limbaj programare C++ sau C#, cunoștințe de matematică de nivel de liceu - profil real, cunoștințe minime de engleză
4.2 de competențe	• Elaborarea unor programe în C++ sau C#.

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	• Tablă, Proiector
5.2 de desfășurare a seminarului/ laboratorului/ proiectului	• Calculatoare, medii de programare C# - Microsoft Visual Studio

6. Competențe specifice acumulate

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt **Cunoștințe** R.Î.1.1. Absolventul poate să scrie cod clar și bine structurat în limbaje de programare specifice. R.Î.1.2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale. **Aptitudini** R.Î.1.3. Absolventul poate să utilizeze biblioteci și framework-uri pentru îmbunătățirea performanțelor și funcționalitatea aplicațiilor software. R.Î.1.4. Absolventul poate să dezvolte aplicații complexe cu cerințele propuse de către utilizatori. **Responsabilitate și autonomie** R.Î.1.5. Absolventul este capabil să organizeze, planifice și supravegheze dezvoltarea aplicațiilor și a cadrelor necesare pentru a crea un produs software complet, de la etapele inițiale de planificare până la testarea finală. CP4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale **Cunoștințe** R.Î.4.1. Absolventul poate să verifice corespondența dintre capacitățile, corectitudinea și eficienta unui algoritm sau a unui sistem intenționat și anumite specificații formale. R.Î.4.2. Absolventul poate să utilizeze modele formale și teorii matematice pentru a rezolva probleme din diverse domenii ale informaticii. **Aptitudini** R.Î.4.3. Absolventul poate să utilizeze concepte și tehnici matematice pentru a analiza și proiecta algoritmi și structuri de date folosind tehnologii specifice. R.Î.4.4. Absolventul poate să utilizeze modele formale pentru a proiecta sisteme informatice sigure și fiabile. R.Î.4.5. Absolventul poate să utilizeze modele formale pentru a verifica corectitudinea programelor. **Responsabilitate și autonomie** R.Î.4.6. Studentul/absolventul construiește etic și responsabil soluții IT sigure și scalabile, colaborând cu specialiști din domenii conexe. R.Î.4.7. Studentul/absolventul dezvoltă un mediu colaborativ și își asumă responsabilitatea pentru succesul livrării proiectelor la timp și conform cerințelor. Studentul/absolventul organizează echipe tehnice și gestionează ciclul de viață al proiectelor software..

Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. R.Î.1.3. Absolventul are capacitatea de a acționa în mod responsabil și etic în activitățile didactice și de cercetare, demonstrând integritate și respect față de colegi, studenți și comunitatea academică. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite. CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. R.Î.2.3. Absolventul poate aplica tehnici de comunicare empatică, ascultând activ și răspunzând adecvat nevoilor și preocupărilor colegilor, contribuind astfel la crearea unui mediu de lucru armonios. R.Î.2.4. Absolventul poate adapta stilul de comunicare în funcție de contextul cultural și diversitatea grupului, asigurându-se că mesajul este transmis eficient și este bine înțeles de toți participanții. CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î.3.1. Absolventul va utiliza metode și tehnici eficiente de învățare pentru a-și îmbunătăți performanța academică. R.Î.3.2. Absolventul va utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis. R.Î.3.3. Absolventul poate evalua critic sursele de informație, identificând cele mai relevante și credibile materiale pentru a sprijini deciziile informate și fundamentate.

7. Obiectivele disciplinei (reieșind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	• Cunoașterea și înțelegerea unor elemente de bază din procesarea imaginilor digitale și adaptarea lor la probleme reale. • Dezvoltarea capacității studenților de a căuta documentație potrivită și de a aplica cunoștințele teoretice în practică
7.2 Obiectivele specifice	• Dezvoltarea capacității studenților de a aborda probleme de procesare de imagine, de a implementa algoritmii de bază. • Dezvoltarea abilității studenților de a interpreta și evalua rezultatele produse de propriile programe, de a adapta metodele generale la probleme concrete.

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
1. Introducere, etapele principale ale procesării imaginilor, discretizarea datelor analoge, reprezentarea imaginilor, relații între pixeli. Spații de culoare.	Expunere Dialog Explicație	2	
2. Îmbunătățirea imaginilor - operații punctuale 2.1 Binarizare		2	

2.2 Modificare luminozitate / contrast 2.3. Histograma	Dezbatere Informare		
3. Segmentare de bază. Thresholding.		2	
4. Filtrare în domeniul spațial: 4.1 Reducerea zgomotului - filtrare trece-jos (filtrul medie aritmetică, filtrul gaussian, filtrul median) 4.2 Detectarea de contur - filtrare trece-sus (margini, modelare matematică, filtrul Sobel, operatorul Canny, operatorul Marr-Hildreth) 4.3 Filtrare avansata		6	
5. Operații morfologice (dilatare, erodare, opening, closing, componente conexe)		2	
6. Transformări geometrice 5.1 Interpolare 5.2 Rotație, scalare, transformări proiective		2	
7. Elemente de segmentare: 7.1 Segmentarea pe baza histogramei - thresholding 7.2 Segmentare bazată pe informație de contur - Transformarea Hough, operatorul Harris 7.3. Segmentare bazată pe regiuni		4	
8. Reprezentare și descriere. Texturi		2	
9. Tehnici în domeniul frecvență 9.1 Transformata Fourier 9.2 Transformata cos		4	
10. Sinteză și recapitulare pentru examen	Dialog.	2	

Bibliografie

1. Gonzalez R.C., Woods R.E., „Digital Image Processing" 4th Edition, Pearson 2018
2. Burger W., Burge M. - Digital Image Processing - 2nd edition, Springer 2016
3. Plajer I. C., Fieraru C.G, Băicoianu A., "Procesarea imaginilor digitale ", Curs pentru învățământul la distanță 2024, https://ebooks.unitbv.ro/produs/procesarea-imaginilor-digitale-curs-pentru-invatamant-la-distanta-anul-ii-sem-ii/

8.2 Laborator	Metode de predare-învățare	Număr de ore	Observații
1. Instalarea și familiarizarea cu programul în care se va lucra la laborator	Aplicații / programare / interpretarea și evaluarea rezultatelor	2	La fiecare capitol de curs, studenții vor primi câte o temă individuală. Fiecare temă are un punctaj aferent. Temele trebuie predate la termenele specificate. Pentru susținerea examenului scris este necesară obținerea unei
2. Implementarea operatorilor punctuali.		2	
3. Implementarea și testarea algoritmilor de thresholding.		2	
4. Implementarea și testarea filtrelor trece-jos și trece-sus		4	
5. Implementarea și testarea operatorilor morfologici.		4	

6. Implementarea și testarea transformărilor geometrice.		4	note minime de 5 la laborator.
7. Implementarea și testarea algoritmilor de segmentare prezentați la curs		6	
8. Evaluare finală		4	

Bibliografie:

1. Gonzalez R.C., Woods R.E., „Digital Image Processing" 4th Edition, Pearson 2018
2. W. Burger, M. Burge - Principles of Digital Image Processing. Core Algorithms
3. Ioannis Pitas, „Digital Image Processing Algorithms", Prentice Hall 1993
4. Pratt K.W., *Digital Image Processing* third edition, John Wiley & Sons, 2001
5. Ivanovici L.M, *Procesarea Imaginilor*, îndrumar de laborator, Universitatea Transilvania 2006
6. Plajer I. C., Fieraru C.G, Băicoianu A., "Procesarea imaginilor digitale ", Curs pentru învățământul la distanță 2024, https://ebooks.unitbv.ro/produs/procesarea-imaginilor-digitale-curs-pentru-invatamant-la-distanta-anul-ii-sem-ii/

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunităților epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

Fișa disciplinei a fost elaborată după consultări cu partenerii din mediul academic și economic ai Facultății.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Atingerea obiectivelor educaționale ale disciplinei	Examen scris	50 %
	Formarea competențelor vizate prin conținutul disciplinei		
10.5 Laborator	Numărul teme realizate precum și gradul de dificultate al acestora.	Evaluarea pe parcurs cu punctaj.	50%
10.6 Standard minim de performanță			

Conținuturi:

Pentru nota la laborator:

- predarea la termen a temelor de laborator elaborate individual, cumulând un total de minim 5 puncte.

Pentru nota 5 la examenul scris

- operatori punctuali: binarizare, operatori liniari pe porțiuni, histograma (definiție și calcul).
- filtre trece-jos și trece sus: proprietăți filtre, măști de filtrare, aplicarea pe imagini
- operatori morfologici: erodare, dilatare, opening și closing (mod de calcul).
- imagini color: modelul RGB.

Observație: alternativ examenul poate fi promovat prin realizarea unui proiect mai complex de procesare de imagine in echipe de maxim 2 persoane. Tematica de proiect se stabilește in mod individual împreună cu cadrul didactic, iar proiectul de prezintă în ultima săptămână din semestru.

Restricții:

Pentru promovarea examenului sunt necesare o notă de minim 5 la laborator și media dintre nota de la examenul scris și cea de la laborator să fie minim 5.

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. dr. Gabriel Stan	Director de departament Conf. dr. Nicușor Minculete
Titular curs Lect. dr. Ioana Cristina Plajer	Titulari laborator Lect. dr. Ioana Cristina Plajer Drd. Cristian George Fieraru

Notă:

1) Domeniul de studii – se alege una din variantele: Licență / Masterat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor / programelor de studii universitare în vigoare).

2) Ciclul de studii – se alege una din variantele: Licență / Masterat.

3) Regimul disciplinei (conținut); se alege una din variantele: DF (disciplină fundamentală) / DD (disciplină din domeniu) / DS (disciplină de specialitate) / DC (disciplină complementară) - pentru nivelul de licență ; DAP (disciplină de aprofundare) / DSI (disciplină de sinteză) / DCA (disciplină de cunoaștere avansată) - pentru nivelul de masterat.

4) Regimul disciplinei (obligativitate) – se alege una din variantele: DI (disciplină impusă) / DO (disciplină opțională) / DFac (disciplină facultativă).

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice și studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii de .....[1)]	Informatică
1.5 Ciclul de studii[2)]	Informatică
1.6 Programul de studii/ Calificarea	Informatică Aplicată / Licențiat în informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Procese stohastice și aplicații							
2.2 Titularul activităților de curs				Prof. Dr. Mihai PASCU				
2.3 Titularul activităților de seminar				Prof. Dr. Mihai PASCU				
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[4)]	DO

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

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar/ laborator/ proiect	0/28/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie şi notițe	30
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	20
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	24
Tutoriat	10
Examinări	10
Alte activități.....	-

3.7 Total ore de activitate a studentului	94
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	-
4.2 de competențe	-

5. Condiții (acolo unde este cazul)

5.1 de desfăşurare a cursului	• Sala de curs să aibă număr suficient de locuri, şi să fie dotată (de preferat) cu videoproiector şi ecran de proiectie.
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	• Sala de seminar să aibă număr suficient de locuri, şi să fie dotată (de preferat) cu videoproiector şi ecran de proiectie.

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențe profesionale	CP4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale **Cunoștințe** R.Î.4.2. Absolventul poate să utilizeze modele formale și teorii matematice pentru a rezolva probleme din diverse domenii ale informaticii. **Aptitudini** R.Î.4.3. Absolventul poate să utilizeze concepte și tehnici matematice pentru a analiza și proiecta algoritmi și structuri de date folosind tehnologii specifice.
Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. R.Î.1.3. Absolventul are capacitatea de a acționa în mod responsabil și etic în activitățile didactice și de cercetare, demonstrând integritate și respect față de colegi, studenți și comunitatea academică. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite. CT2 Desfăşurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. R.Î.2.3. Absolventul poate aplica tehnici de comunicare empatică, ascultând activ și răspunzând adecvat nevoilor și preocupărilor colegilor, contribuind astfel la crearea unui mediu de lucru armonios. R.Î.2.4. Absolventul poate adapta stilul de comunicare în funcție de contextul cultural și diversitatea grupului, asigurându-se că mesajul este transmis eficient și este bine înțeles de toți participanții. CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoştințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î.3.1. Absolventul va utiliza metode și tehnici eficiente de învățare pentru a-și îmbunătăți performanța academică. R.Î.3.2. Absolventul va utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis. R.Î.3.3. Absolventul poate evalua critic sursele de informație, identificând cele mai relevante și credibile materiale pentru a sprijini deciziile informate și fundamentate.

7. Obiectivele disciplinei (reieşind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	• Să ofere studenților bagajul de cunoștințe necesar pentru înțelegerea problemelor și a posibilelor aplicații ale proceselor stochastice.
7.2 Obiectivele specifice	• Studentul să înțeleagă notiunile fundamentale din teoria Proceselor Stochastice (aşteptare conditionată, martingale, timpi de oprire, integrală

	stochastică), proprietătile acestora, precum şi diversele aplicatii ale acestora. • Dobândirea unor cunoştiinte ştiintifice solide în Teoria probabilităţilor şi a Proceselor stochastice, ca baza a dezvoltarilor ulterioare • Explicarea şi interpretarea corectă a conceptelor matematice fundamentale în Probabilităţi şi Procese stochastice, şi realizarea de interconxiuni între acestea • Aplicarea corectă a metodelor şi principiilor de bază ale Probabilităţilor şi a Proceselor stochastice în rezolvarea unor probleme concrete • Capacitatea de a selecta metodele şi tehnicile probabiliste adecvate rezolvării unor probleme concrete • Cunoaşterea principiilor şi a metodelor generale de modelare matematică • Dobândirea capacită ii de constructie şi explicare a modelelor matematice • Capacitatea de a construi modele stochastice adecvate unei situatii noi pe baza celor studiate • Capacitatea de analiză a diferitelor modele prin metode stochastice • Obtinerea abilităţilor de aplicare a cunoştiintelor de matematică şi a metodelor computationale în Procese stochastice • Aplicarea în situatii practice a tehnicilor proceselor stochastice • Capacitatea de rezolvare a problemelor prin utilizarea aparatului teoretic si a metodelor de calcul ale proceselor stochastice • Cunoaşterea aprofundată a stadiului actual al cunoaşterii în domeniul Proceselor stochastice • Identificarea sarcinilor şi activităţătilor necesare pentru abordarea unei teme de cercetare • Analiza comparativa a diferitelor strategii de desfăşurarea a activităţii de cercetare

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
1. Spatiu de probabilitate (sigma-algebră, măsura de probabilitate, spatiu de probabilitate, variabile aleatoare. Exemple şi proprietăţi)	Clasic	2	
2. Aşteptare conditionată în raport cu o sigma-algebră (filtratie, aşteptare conditionată. Exemple şi proprietăţi)		4	
3. Martingale (martingale / submartingale / supermartingale. Exemple şi proprietăţi). Timpi de oprire (definitii, exemple, proprietăţi)		4	
4. Teorema Doob a timpului optional de oprire (cazul martingalelor discrete/continue, a timpului de oprire mărginit. Generalizări)		3	
5. Transformări martingale (definitii, exemple, proprietăţi)		3	
6. Inegalităţi şi teoreme de convergentă (inegalităţi maximale, teorema		3	

inegalitătilor de traversări crescătoare / descrescătoare, aplicatie la convergentă a submartingalelor)			
7. Mişcarea Browniană. Definitie, constructie, proprietati.		2	
8. Integrala stochastică (constructie, proprietăti)		4	
9. Formula Itô (cazul 1-dimensional, extensii la cazul general)		3	

Bibliografie

1. R. F. Bass, Probabilistic techniques in Analysis, Springer-Verlag, New York, 1995.
2. R. F. Bass, Diffusions and elliptic operators, Springer-Verlag, New York, 1997.
3. M. N. Pascu, Brownian motion and Applications, Transilvania University Press, Braşov, 2006.
4. M. N. Pascu, Calcul stochastic, mişcare Browniană şi aplica ii, Editura Universității Transilvania Braşov, Braşov, 2010.
5. M. N. Pascu, Notițe de curs (disponibile pe pagina Elearning a cursului)

8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore	Observații
1. Spatiu de probabilitate (sigma-algebră, măsura de probabilitate, spatiu de probabilitate, variabile aleatoare. Exemple şi proprietăti)	Învătare prin exercitii (aplicatii)	2	
2. Aşteptare conditionată în raport cu o sigma-algebră (filtratie, aşteptare condi ionată. Exemple şi proprietăti)		2	
3. Martingale (martingale/submartingale/supermartingale. Exemple şi proprietăti). Timpi de oprire (definitii, exemple, proprietăti)		2	
4. Teorema Doob a timpului optional de oprire (cazul martingalelor discrete/continue, a timpului de oprire mărginit. Generalizări)		2	
5. Transformări martingale (definitii, exemple, proprietă i)		2	
6. Inegalităti şi teoreme de convergen ă (inegalităti maximale, teorema inegalităților de traversări crescătoare/descrescătoare, aplicatie la convergentă a submartingalelor)		2	
7. Mişcarea Browniană. Definitie, constructie, proprietati.	Învătare prin exercitii (aplicatii)	2	
8. Integrala stochastică (constructie, proprietăti)		2	
9. Formula Itô (cazul 1-dimensional, extensii la cazul general)		2	
10. Rezolvarea probabilistă a problemei Dirichlet pentru ecuatia lui Laplace		2	

11. Rezolvarea probabilistă a problemei Dirichlet pentru ecuatia lui Poisson		2	
12. Rezolvarea probabilistă a ecuatiei căldurii (conditii Dirichlet / Neumann pe frontiera domeniului)		2	
13. Formula Feynman-Kac		2	
14. Formula Black-Merton-Scholes		2	

Bibliografie

1. R. F. Bass, Probabilistic techniques in Analysis, Springer-Verlag, New York, 1995.
2. R. F. Bass, Diffusions and elliptic operators, Springer-Verlag, New York, 1997.
3. M. N. Pascu, Brownian motion and Applications, Transilvania University Press, Braşov, 2006.
4. M. N. Pascu, Calcul stochastic, mişcare Browniană şi aplica ii, Editura Universității Transilvania Braşov, Braşov, 2010.
5. M. N. Pascu, Notițe de curs (disponibile pe pagina Elearning a cursului)

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, ale asociaților profesionale şi ale angajatorilor reprezentativi din domeniul aferent programului

Processele stochastice sunt foarte folositoare în studiul situatiilor guvernate de factori aleatori. Cum în majoritatea problemelor din domeniul economic/privat apar astfel de factori, este de aşteptat ca acest curs să fie un plus pentru stundetii care se orientează către sectorul privat

10. Evaluare

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Nota obtinută în examen	Examen scris	50%
10.5 Seminar/ laborator/ proiect	Nota obtinută în examen	Examen scris	50%
10.6 Standard minim de performanță			
• Nota minimă pentru promovare este 5 (cinci)			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Conf. dr. Ion Gabriel Stan **Decan**	**Conf. dr. Nicușor Minculete** **Director de departament**
Prof. Dr. Mihai PASCU **Titular de curs**	**Prof. Dr. Mihai PASCU** **Titular de laborator/seminar**

Notă:

[1] Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

[2] Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

[3] Regimul disciplinei (conținut) - se alege una din variantele**: DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

[4] Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

[5] Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii de licență[1)]	Informatică
1.5 Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	INFORMATICĂ APLICATĂ / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Introducere în calcul cuantic							
2.2 Titularul activităților de curs				Sorana Catrina (Harman international industries)				
2.3 Titularul activităților de seminar/ laborator/proiect				Sorana Catrina				
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[4)]	DO

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

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar/ laborator/ proiect	14/14/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie și notițe	33
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	10
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri	40
Tutoriat	7
Examinări	4
Alte activități.....	0

3.7 Total ore de activitate a studentului	94
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Cunoștințe de bază de algebră liniară și programare.
4.2 de competențe	• Competențe elementare de programare și de rezolvare a problemelor simple de algebră liniară

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	• Tablă, Proiector
5.2 de desfășurare a seminarului/ laboratorului/ proiectului	• Internet, mediu de dezvoltare Python capabil de gestionarea diferitelor biblioteci Python.

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt **Cunoștințe** R.Î.1.1. Absolventul poate să scrie cod clar și bine structurat în limbaje de programare specifice. R.Î.1.2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale. **Aptitudini** R.Î.1.3. Absolventul poate să utilizeze biblioteci și framework-uri pentru îmbunătățirea performanțelor și funcționalitatea aplicațiilor software. **Responsabilitate și autonomie** R.Î.1.5. Absolventul este capabil să organizeze, planifice și supravegheze dezvoltarea aplicațiilor și a cadrelor necesare pentru a crea un produs software complet, de la etapele inițiale de planificare până la testarea finală. CP3. Utilizarea instrumentelor informatice în context interdisciplinar **Cunoștințe** R.Î.3.1. Absolventul poate utiliza instrumente informatice (folosind tehnologii specifice) pentru a colecta, analiza și interpreta date din diferite domenii de activitate. R.Î.3.2. Absolventul poate evalua și selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare, având abilitatea de a analiza nevoile specifice ale unui proiect și de a alege soluții tehnologice eficiente. **Aptitudini** R.Î.3.3. Absolventul poate implementa metode de testare și evaluare a software-ului, inclusiv testarea interactivă și feedback-ul utilizatorilor din diverse discipline, pentru a asigura calitatea și funcționalitatea aplicațiilor create. R.Î.3.4. Studentul/absolventul dezvoltă interesul pentru literatura de specialitate națională și internațională, participarea la cursuri specializate, conferințe, simpozioane, etc. pentru dezvoltarea profesională şi personală continuă și adaptarea eficientă la noile descoperiri ştiin ifice. **Responsabilitate și autonomie** R.Î.3.5. Absolventul poate să comunice și să colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice, dezvoltând competențe de lucru în echipă și abilități de gestionare a proiectelor interdisciplinare. CP4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale **Cunoștințe** R.Î.4.1. Absolventul poate să verifice corespondența dintre capacitățile, corectitudinea și eficienta unui algoritm sau a unui sistem intenționat și anumite specificații formale. R.Î.4.2. Absolventul poate să utilizeze modele formale și teorii matematice pentru a rezolva probleme din diverse domenii ale informaticii. **Aptitudini** R.Î.4.3. Absolventul poate să utilizeze concepte și tehnici matematice pentru a analiza și proiecta algoritmi și structuri de date folosind tehnologii specifice. R.Î.4.4. Absolventul poate să utilizeze modele formale pentru a proiecta sisteme informatice sigure și fiabile. R.Î.4.5. Absolventul poate să utilizeze modele formale pentru a verifica corectitudinea programelor. **Responsabilitate și autonomie** R.Î.4.6. Studentul/absolventul construiește etic și responsabil soluții IT sigure și scalabile, colaborând cu specialiști din domenii conexe. R.Î.4.7. Studentul/absolventul dezvoltă un mediu colaborativ și își asumă responsabilitatea pentru succesul livrării proiectelor la timp și conform cerințelor. Studentul/absolventul organizează echipe tehnice și gestionează ciclul de viață al proiectelor software.gestionează ciclul de viață al proiectelor software.

Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. R.Î.1.3. Absolventul are capacitatea de a acționa în mod responsabil și etic în activitățile didactice și de cercetare, demonstrând integritate și respect față de colegi, studenți și comunitatea academică. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. R.Î.2.3. Absolventul poate aplica tehnici de comunicare empatică, ascultând activ și răspunzând adecvat nevoilor și preocupărilor colegilor, contribuind astfel la crearea unui mediu de lucru armonios. R.Î.2.4. Absolventul poate adapta stilul de comunicare în funcție de contextul cultural și diversitatea grupului, asigurându-se că mesajul este transmis eficient și este bine înțeles de toți participanții. CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î.3.1. Absolventul va utiliza metode și tehnici eficiente de învățare pentru a-și îmbunătăți performanța academică. R.Î.3.2. Absolventul va utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis. R.Î.3.3. Absolventul poate evalua critic sursele de informație, identificând cele mai relevante și credibile materiale pentru a sprijini deciziile informate și fundamentate.

7. Obiectivele disciplinei (reieșind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	Însușirea noțiunilor elementare ce stau la baza programării cuantice, precum superpoziția și entanglement-ul („*încâlcirea*"), și familiarizarea cu mediul de dezvoltare a algoritmilor cuantici.
7.2 Obiectivele specifice	• Familiarizarea cu suportul matematic necesar pentru înțelegerea modului de funcționare a algoritmilor cuantici • Familiarizarea cu diferitele sub-domenii din domeniul programării cuantice și identificarea instrumentelor necesare și rolul acestora • Dezvoltarea unor programe simple ce pun în practică noțiunile teoretice învățate

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
Recapitularea conceptelor de algebră liniară și statistică matematică	Expunere	6	

Introducere în mecanica cuantică (pentru calcul cuantic)	Dialog Explicație Dezbatere Informare Dialog	4	
Introducere în algoritmi cuantici		8	
Hardware Cuantic		4	
Rețele cuantice: Distribuirea cheilor cuantice (QKD)		6	
Bibliografie - Nielsen, M. A., & Chuang, I. L. (2010). *Quantum computation and quantum information* (10th anniversary ed.). Cambridge University Press - Griffiths, D. J. (2018). *Introduction to quantum mechanics* (3rd ed.). Cambridge University Press. - Sakurai, J. J., & Napolitano, J. (2017). *Modern quantum mechanics* (2nd ed., revised). Cambridge University Press.			
8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore	Observații
Familiarizarea cu mediul de lucru și cu bibliotecile necesare	Exerciții. Rezolvare exerciții. Exemplificarea noțiunilor teoretice. Lucrări practice Discutarea algoritmilor. Verificări și corectări de programe.	4	
Aplicarea conceptelor învățate la curs creând diferite programe pentru rezolvarea unor probleme		4	
Implementarea algoritmilor cuantici prezentați folosind instrumentele de programare specifice		8	
Rularea programelor scrise pe hardware cuantic, evidențiind pașii ce trebuie făcuți și elementele ce trebuie luate în considerare atunci când se rulează un program cuantic pe hardware.		4	
Implementarea protocoalelor de comunicare discutate la curs		4	
Prezentări de teme și proiecte		4	
Bibliografie - Nielsen, M. A., & Chuang, I. L. (2010). *Quantum computation and quantum information* (10th anniversary ed.). Cambridge University Press - Griffiths, D. J. (2018). *Introduction to quantum mechanics* (3rd ed.). Cambridge University Press. - IBM Quantum. (n.d.). *IBM Quantum Learning*, https://learning.quantum.ibm.com/			

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunităților epistemice, ale asociațiilor profesionale și ale angajatorilor reprezentativi din domeniul aferent programului

Fișa disciplinei a fost elaborată după consultări cu partenerii din mediul academic și economic ai Facultății.

10. Evaluare

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Atingerea obiectivelor educaționale ale disciplinei	Examen scris și practic (rezolvarea problemelor matematice și implementarea rezolvărilor unor probleme) din	50%
	Formarea competențelor vizate prin conținutul		

	disciplinei	capitolele prezentate la curs.	
10.5 Seminar/ laborator/ proiect	Implementarea aplicațiilor formulate, conform cu baremul pentru fiecare temă.	Evaluare continuă prin activitățile specifice fiecărui capitol și prezentarea proiectului final.	50%
10.6 Standard minim de performanță			
Conținuturi: Pentru nota la **laborator**: Implementarea unui algoritm cuantic simplu și interpretarea corectă a rezultatelor. Pentru nota la **examenul scris** Familiaritate cu noțiunile de bază ale fundamentului matematic și ale principiilor programării cuantice. Rezolvarea unor probleme simple de calcul cuantic. **Restricții**: Pentru promovarea examenului sunt necesare o notă de minim 5 la laborator și o notă de minim 5 la examenul scris.			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf.dr. Gabriel Stan	Director de departament Conf. dr. Nicușor Minculete
Titulari curs Sorana Catrina	Titulari laborator / seminar Sorana Catrina

Notă:

1) Domeniul de studii – se alege una din variantele: Licență / Masterat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor / programelor de studii universitare în vigoare).

2) Ciclul de studii – se alege una din variantele: Licență / Masterat/ Doctorat.

3) Regimul disciplinei (conținut); se alege una din variantele: DF (disciplină fundamentală) / DD (disciplină din domeniu) / DS (disciplină de specialitate) / DC (disciplină complementară) - pentru nivelul de licență ; DAP (disciplină de aprofundare) / DSI (disciplină de sinteză) / DCA (disciplină de cunoaștere avansată) - pentru nivelul de masterat.

4) Regimul disciplinei (obligativitate) – se alege una din variantele: DI (disciplină impusă) / DO (disciplină opțională) / DFac (disciplină facultativă).

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice și studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Braşov
1.2 Facultatea	Matematică şi Informatică
1.3 Departamentul	Matematică şi Informatică
1.4 Domeniul de studii de licență [1)]	Informatică
1.5 Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	Informatică aplicată // Licențiat în informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Inginerie soft							
2.2 Titularul activităților de curs				Conf. dr. Cârstea Claudia				
2.3 Titularul activităților de seminar/ laborator/ proiect				Conf. dr. Cârstea Claudia				
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar/ laborator/ proiect	0/28/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie şi notițe	40
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	20
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	15
Tutoriat	10
Examinări	9
Alte activități.....	-

3.7 Total ore de studiu individual	94
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• POO, Teoria sistemelor, Programare Java
4.2 de competențe	• Abilități de programare într-un limbaj orientat-obiect

5. Condiții (acolo unde este cazul)

5.1 de desfăşurare a cursului	- videoproiector
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	Calculatoare pe care sa fie instalate sisteme de operare din familia Windows şi Visual Paradigm for UML

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențe profesionale	CP2. Gestionarea ciclului de viață al aplicațiilor informatice **Cunoștințe** R.Î.2.1. Absolventul poate să parcurgă ciclul de viață al unui sistem informatic, începând cu ancheta preliminară și terminând cu exploatarea și menținerea sistemului. R.Î.2.2. Absolventul poate să identifice și să evalueze riscurile asociate cu fiecare fază a ciclului de viață. **Aptitudini** R.Î.2.3. Absolventul poate să proiecteze soluții software eficiente și scalabile pentru fiecare fază a ciclului de viață. R.Î.2.4. Absolventul poate să implementeze și să testeze componente software conform specificațiilor funcționale și non-funcționale. R.Î.2.5. Absolventul poate să monitorizeze și să optimizeze performanța sistemului în faza de exploatare. R.Î.2.6. Absolventul poate să aplice măsuri de securitate pentru protejarea datelor pe tot parcursul ciclului de viață. R.Î.2.7. Absolventul poate realiza documentație tehnică și manuale de utilizare pentru aplicațiile software, facilitând astfel utilizarea și întreținerea acestora de către specialiști din diverse domenii. **Responsabilitate și autonomie** R.Î.2.8. Absolventul poate să lucreze în echipă pentru a parcurge ciclul de viață al unui sistem informatic. CP3. Utilizarea instrumentelor informatice în context interdisciplinar **Cunoștințe** R.Î.3.1. Absolventul poate utiliza instrumente informatice (folosind tehnologii specifice) pentru a colecta, analiza și interpreta date din diferite domenii de activitate. R.Î.3.2. Absolventul poate evalua și selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare, având abilitatea de a analiza nevoile specifice ale unui proiect și de a alege soluții tehnologice eficiente. **Aptitudini** R.Î.3.3. Absolventul poate implementa metode de testare și evaluare a software-ului, inclusiv testarea interactivă și feedback-ul utilizatorilor din diverse discipline, pentru a asigura calitatea și funcționalitatea aplicațiilor create. **Responsabilitate și autonomie** R.Î.3.5. Absolventul poate să comunice și să colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice, dezvoltând competențe de lucru în echipă și abilități de gestionare a proiectelor interdisciplinare.
Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională. CT2 Desfăşurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoştințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională.

7. Obiectivele disciplinei (reieşind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	• Cunoașterea și aplicarea unor concepte, principii și tehnici de dezvoltare a sistemelor soft • Formarea si dezvoltarea abilităților de utilizare a tehnicilor suport pentru ingineria softului • Dobândirea de cunoştinte specifice metodologiilor Scrum, Kanban, Extreme Programming etc • Înțelegerea modului de aplicare a practicilor Agile.
7.2 Obiectivele specifice	• Conştientizarea avantajelor abordării metodologice în industria softului • Cunoașterea importantei modelelor de dezvoltare si aplicarea selectiva a acestora • Cunoașterea limbajelor de modelare • Formarea si dezvoltarea deprinderilor de utilizare a limbajelor de modelare in analiza si proiectarea sistemelor informatice complexe • Determinarea aspectelor care fac metodologiile Agile superioare metodologiilor clasice pentru proiectele software • Determinarea punctelor tari şi a punctelor slabe ale fiecăreia dintre practicile Agile utilizate astăzi • Determinarea ciclului de viată al unui proiect software într-un context Agile • Cunoașterea etapelor ciclului de viată al softului și a modelelor de procese soft • Însușirea aspectelor de bază legate de gestiunea softului • Familiarizarea cu unele dintre metodologiile de dezvoltare, tradiționale sau agile • Înțelegerea conceptelor legate de modelarea softului • Cunoașterea și aplicarea tehnicilor de dezvoltare a softului pe baza modelelor, precum și a diferitor metode de testare • Familiarizarea cu limbajul UML • Abilitatea de a utiliza instrumente CASE

8. Conținuturi

8.1 Curs	**Metode de predare**	**Număr de ore**	**Observații**
1. Elemente de teoria generală a sistemelor din perspectiva ingineriei sistemelor soft. Noțiunea de metodologie • Modele de dezvoltare • Limbaje de modelare	prelegerea, explicația, demonstrația, problematizarea, exercițiul	2	
2. Introducere în ingineria sistemelor soft: motivație, definiție, concepte, activități. Ciclul de viată al softului. Metodologii de dezvoltare a sistemelor soft.	prelegerea, explicația, demonstrația, problematizarea, exercițiul	2	
3. Sisteme informaționale. Fluxuri informaționale. Sisteme informaționale cu componente automatizate.	prelegerea, explicația,	2	
4. Limbajul de modelare UML	demonstrația, problematizarea,	2	
5. Tipuri de diagrame UML	exercițiul	2	
6. Diagrame de stare Tools pentru asistarea dezvoltării UML a soluției unei	prelegerea, explicația,	2	

probleme.			
7. Prezentarea tool-lui Visual Paradigm for UML	demonstrația, problematizarea,	2	
8. Diagrame de comunicare Tools pentru asistarea dezvoltării UML a soluției unei probleme. Prezentarea tool-lui Visual Paradigm for UML	prelegerea, explicația, demonstrația, problematizarea, exercițiul	2	
9. Diagrame de activități. Diagrame de interacțiune.	prelegerea, explicația,	2	
10. Diagrame de pachete.	demonstrația, problematizarea,	2	
11. Diagrame de desfăşurare.	prelegerea, explicația,	2	
12. Diagrame de componente.	demonstrația, problematizarea,	2	
13. Proiectarea unui sistem informatic integrat pentru managementul resurselor umane	Exercițiul	4	

Bibliografie

- C., Cârstea, Note de curs, Ingineria sistemelor Software, 2018-2019
- C., Cârstea, Ghid Inginerie Software, ed. Universității Transilvania, Braşov, 2017
- Jeff Langr, Tim Ottinger - Agile in a Flash: Speed-Learning Agile Software Development, Pragmatic Bookshelf, 2011
- Esther Derbt, Diana Larsen - Agile Retrospectives: Making Good Teams Great, Pragmatic Bookshelf, 2006
- Thomas Stober, Uve Hansmann - Agile Software Development, Best Practices for Large Software Development Projects, Springer 2010
- Mike Cohn - Succeeding with Agile Software Development using Scrum, Addison-Weslet, 2010

8.2 Seminar/ laborator/ proiect	**Metode de predare-învățare**	**Număr de ore**	**Observații**
Utilizarea tool-ului Visual Paradigm for UML pentru analiza şi design-ul soluției unui sistem soft. Elaborarea unui proiect în echipă. Tema 1: Exemplificarea utilității abordării sistemice în cadrul proiectelor IT; Exemplificarea utilității abordării metodologice în cadrul proiectelor IT	Sesiuni demo de utilizare a tool-ului Visual Paradigm; discutarea specificațiilor aferente diferitelor faze ale evoluției proiectelor Expunere, explicare, exemplificare, dezbatere	4	
Tema 2: Exemplificarea definirii obiectivelor şi a cerințelor aferente unui proiect IT. Reprezentare cu suport VP UML.	Sesiuni demo de utilizare a tool-ului Visual Paradigm; discutarea specificațiilor aferente diferitelor faze ale evoluției proiectelor Expunere, explicare, exemplificare, dezbatere	2	
Tema 3: Exemplificarea analizei sistemului informațional existent cu suport VP UML.	Sesiuni demo de utilizare a tool-ului Visual Paradigm; discutarea specificațiilor aferente diferitelor faze ale	4	

	evoluției proiectelor Expunere, explicare, exemplificare, dezbatere		
Tema 4: Diagrame cazuri de utilizare	Sesiuni demo de utilizare a tool-ului Visual Paradigm; discutarea specificațiilor aferente diferitelor faze ale evoluției proiectelor Expunere, explicare, exemplificare, dezbatere	2	
Tema 5: Exemplificarea analizei sistemului informațional existent cu suport VP UML.	Sesiuni demo de utilizare a tool-ului Visual Paradigm; discutarea specificațiilor aferente diferitelor faze ale evoluției proiectelor Expunere, explicare, exemplificare, dezbatere	4	
Tema 6: Diagrame de clase	Sesiuni demo de utilizare a tool-ului Visual Paradigm; discutarea specificatiilor aferente diferitelor faze ale evolutiei proiectelor Expunere, explicare, exemplificare, dezbatere	4	
Tema 7: Exemplificarea capturării aspectelor structurale în cadrul unui proiect IT, cu suport VP UML.	Sesiuni demo de utilizare a tool-ului Visual Paradigm; discutarea specificațiilor aferente diferitelor faze ale evolutiei proiectelor Expunere, explicare, exemplificare, dezbatere	2	
Tema 8: Diagrame de stare	Sesiuni demo de utilizare a tool-ului Visual Paradigm; discutarea specificațiilor aferente diferitelor faze ale evoluției proiectelor Expunere, explicare, exemplificare, dezbatere	2	
Tema 9: Exemplificarea capturării aspectelor comportamentale în cadrul unui proiect IT, cu suport VP UML.	Sesiuni demo de utilizare a tool-ului Visual Paradigm; discutarea specificațiilor aferente diferitelor faze ale	2	

	evoluției proiectelor Expunere, explicare, exemplificare, dezbatere		
Tema 10: Diagrame de comportament	Sesiuni demo de utilizare a tool-ului Visual Paradigm; discutarea specificațiilor aferente diferitelor faze ale evoluției proiectelor Expunere, explicare, exemplificare, dezbatere	2	

Bibliografie

- Booch, G., Rumbaugh, J., Jacobson, I., The Unified Modeling Language User Guide - V.2.0, Addison Weslet, 2005.
- Bruegge, B., Dutoit, A., Object-Oriented Software Engineering Using UML, Patterns and Java - 3rd Edition, Prentice Hall, 2009.
- C., Cârstea, Note de curs, Ingineria sistemelor Software, 2018-2019
- C., Cârstea, Ghid Inginerie Software, ed. Universităţii Transilvania, Braşov, 2017
- Esther Derbt, Diana Larsen - Agile Retrospectives: Making Good Teams Great, Pragmatic Bookshelf, 2006
- Fowler, M. et al., Refactoring - Improving the Design of Existing Code, Addison Weslet, 1999.
- Fowler, M., Scott, K., UML Distilled: A Brief Guide to the Standard Object Modeling Language -2nd ed., Addison-Weslet, 1999.
- Gamma, E., Helm, R., Johnson, R., Vlissides, J., Design Patterns, Addison-Weslet, 1996.
- Jeff Langr, Tim Ottinger - Agile in a Flash: Speed-Learning Agile Software Development, Pragmatic Bookshelf, 2011
- Martin, R.C., Agile Software Development: Principles, Patterns, and Practices, Prentice Hall, 2002.
- Mike Cohn - Succeeding with Agile Software Development using Scrum, Addison-Weslet, 2010
- Mike Cohn - User Stories Applied, For Agile Software Development, Addison-Weslet, 2004
- Pressman, R.S., Software Engineering - A Practitioners Approach - 6th ed., McGraw-Hill, 2005.
- Tom Demarco - Waltzing with Bears Managing Risks On Software Projects
- Sommerville, I., Software Engineering - 8th edition, Addison-Weslet, 2006

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, ale asociaților profesionale şi ale angajatorilor reprezentativi din domeniul aferent programului

Conținuturile sunt în concordantă cu problematica monografiilor apărute în ultimii ani în literatura de specialitate.

10. Evaluare

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Se testează capacitatea de a utiliza conceptele prezentate in diverse etape de analiza si proiectare a sistemelor informatice, precum si	Lucrare scrisă – test grila	50 %

	aplicarea acestor noțiuni prin utilizarea standardelor de modelare.		
	Aplicarea noțiunilor	Rezolvarea aplicațiilor propuse în cadrul lucrării scrise – test grila	10 %
10.5 Seminar/ laborator/ proiect	Este evaluata calitatea prezentării referatului si argumentarea temei acestuia. Se evaluează deprinderile de utilizare a instrumentelor de modelare UML.	Proiectarea de aplicații asemănătoare celor făcute în cadrul orelor de laborator Prezentarea studiilor de caz	10 % 10%
	Proiect	Redactarea coerenta şi corectă proiectului, în concordantă cu cerințele specifice.	20 %
10.6 Standard minim de performanță			
• capacitatea de a utiliza conceptele prezentate in diverse etape de analiza si proiectare a sistemelor informatice, precum si aplicarea acestor noțiuni prin utilizarea standardelor de modelare. • calitatea prezentării referatului si argumentarea temei acestuia.			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Conf. dr. Ion Gabriel Stan Decan	Conf. dr. Nicușor Minculete Director de departament
Conf. dr. Claudia Cârstea Titular curs	Conf. dr. Claudia Cârstea Titular laborator

Notă:

[1)] Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

[2)] Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

[3)] Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

[4)] Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

[5)] Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Institu ia de învă ământ superior	Universitatea „Transilvania” din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii	Informatică
1.5 Ciclul de studii[1)]	Licență
1.6 Programul de studii/ Calificarea	INFORMATICĂ APLICATĂ / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Dezvoltarea jocurilor pe calculator cu Unity							
2.2 Titularul activită ilor de curs				Lect. Dr. Ichim Vasile Sebastian				
2.3 Titularul activită ilor de seminar/ laborator/ proiect				Lect. Dr. Ichim Vasile Sebastian				
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Con inut[2)]	DF
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 seminar/ laborator/ proiect	**2**
3.4 Total ore din planul de învă ământ	56	din care: 3.5 curs	28	3.6 seminar/ laborator/ proiect	28

Distribu ia fondului de timp	ore
Studiul după manual, suport de curs, bibliografie şi noti e	40
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	15
Pregătire seminarii/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	15
Tutoriat	10
Examinări	10
Alte activită i	4
3.7 Total ore studiu individual	94
3.8 Total ore pe semestru	150
3.9 Numărul de credite[4)]	5

4. Precondi ii (acolo unde este cazul)

4.1 de curriculum	• Programare Orientată pe Obiecte în C++
4.2 de competen e	• Descrierea adecvată a paradigmelor de programare și a mecanismelor de limbaj specifice, precum și identificarea diferenței de ordin semantic și sintactic.

5. Condi ii (acolo unde este cazul)

5.1 de desfăşurare a cursului	• Sală de curs dotată cu video proiector
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	• Laborator dotat cu calculatoare

6. Competen e specifice acumulate

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt Cunoștințe R.Î.1.1. Absolventul poate să scrie cod clar și bine structurat în limbaje de programare specifice. R.Î.1.2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale. Aptitudini R.Î.1.3. Absolventul poate să utilizeze biblioteci și framework-uri pentru îmbunătățirea performanțelor și funcționalitatea aplicațiilor software. R.Î.1.4. Absolventul poate să dezvolte aplicații complexe cu cerințele propuse de către utilizatori. Responsabilitate și autonomie R.Î.1.5. Absolventul este capabil să organizeze, planifice și supravegheze dezvoltarea aplicațiilor și a cadrelor necesare pentru a crea un produs software complet, de la etapele inițiale de planificare până la testarea finală. CP2. Gestionarea ciclului de viață al aplicațiilor informatice Cunoștințe R.Î.2.1. Absolventul poate să parcurgă ciclul de viață al unui sistem informatic, începând cu ancheta preliminară și terminând cu exploatarea și menținerea sistemului. Aptitudini R.Î.2.3. Absolventul poate să proiecteze soluții software eficiente și scalabile pentru fiecare fază a ciclului de viață. R.Î.2.7. Absolventul poate realiza documentație tehnică și manuale de utilizare pentru aplicațiile software, facilitând astfel utilizarea și întreținerea acestora de către specialiști din diverse domenii. Responsabilitate și autonomie R.Î.2.8. Absolventul poate să lucreze în echipă pentru a parcurge ciclul de viață al unui sistem informatic. CP3. Utilizarea instrumentelor informatice în context interdisciplinar Cunoștințe R.Î.3.1. Absolventul poate utiliza instrumente informatice (folosind tehnologii specifice) pentru a colecta, analiza și interpreta date din diferite domenii de activitate. R.Î.3.2. Absolventul poate evalua și selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare, având abilitatea de a analiza nevoile specifice ale unui proiect și de a alege soluții tehnologice eficiente. Aptitudini R.Î.3.3. Absolventul poate implementa metode de testare și evaluare a software-ului, inclusiv testarea interactivă și feedback-ul utilizatorilor din diverse discipline, pentru a asigura calitatea și funcționalitatea aplicațiilor create. R.Î.3.4. Studentul/absolventul dezvoltă interesul pentru literatura de specialitate națională și internațională, participarea la cursuri specializate, conferințe, simpozioane, etc. pentru dezvoltarea profesională şi personală continuă și adaptarea eficientă la noile descoperiri ştiin ifice. Responsabilitate și autonomie R.Î.3.5. Absolventul poate să comunice și să colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice, dezvoltând competențe de lucru în echipă și abilități de gestionare a proiectelor interdisciplinare.

Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. R.Î.1.3. Absolventul are capacitatea de a acționa în mod responsabil și etic în activitățile didactice și de cercetare, demonstrând integritate și respect față de colegi, studenți și comunitatea academică. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. R.Î.2.3. Absolventul poate aplica tehnici de comunicare empatică, ascultând activ și răspunzând adecvat nevoilor și preocupărilor colegilor, contribuind astfel la crearea unui mediu de lucru armonios. R.Î.2.4. Absolventul poate adapta stilul de comunicare în funcție de contextul cultural și diversitatea grupului, asigurându-se că mesajul este transmis eficient și este bine înțeles de toți participanții. CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î.3.1. Absolventul va utiliza metode și tehnici eficiente de învățare pentru a-și îmbunătăți performanța academică. R.Î.3.2. Absolventul va utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis. R.Î.3.3. Absolventul poate evalua critic sursele de informație, identificând cele mai relevante și credibile materiale pentru a sprijini deciziile informate și fundamentate.

7. Obiectivele disciplinei (reieşind din grila competen elor specifice acumulate)

7.1 Obiectivul general al disciplinei	Dezvoltarea jocurilor 3D folosind Unity și Blender
7.1 Obiectivele specifice	• Familiarizarea interacțiunii dintre interfața grafică și scripturi • Implementarea fizicii și a navigării • Realizarea animațiilor, efectelor sonore, și a sistemelor de particule • Proiectarea interfeței pentru utilizator în Unity

8. Con inuturi

8.1 Curs	Metode de predare	Număr de ore
Manipularea obiectelor	Expunere și discuție liberă	4
Fizica și coliziunea obiectelor	Expunere și discuție liberă	4
Teren și detectarea obstacolelor	Expunere și discuție liberă	4
Animații, sunete și sisteme de particule	Expunere și discuție liberă	4

Materiale, Shadere și instanșiere	Expunere și discuție liberă	4
Texturi, Animator și NavMesh	Expunere și discuție liberă	4
Meniuri și elemente de interfață cu utilizatorul	Expunere și discuție liberă	4
Bibliografie Victor G. Brusca, Advanced Unity Game Development: Build Professional Games with Unity, C#, and Visual Studio, Editura Apress, 2022 Casey Hardman, Game Programming with Unity and C#, A Complete Beginner's Guide, Editura Apress, 2022		
8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore
Manipularea obiectelor	Învățare prin probleme	4
Fizica și coliziunea obiectelor	Învățare prin probleme	4
Teren și detectarea obstacolelor	Învățare prin probleme	4
Animații, sunete și sisteme de particule	Învățare prin probleme	4
Materiale, Shadere și instanșiere	Învățare prin probleme	4
Texturi, Animator și NavMesh	Învățare prin probleme	4
Meniuri și elemente de interfață cu utilizatorul	Învățare prin probleme	4
Bibliografie Victor G. Brusca, Advanced Unity Game Development: Build Professional Games with Unity, C#, and Visual Studio, Editura Apress, 2022 Casey Hardman, Game Programming with Unity and C#, A Complete Beginner's Guide, Editura Apress, 2022		

9. Coroborarea con inuturilor disciplinei cu aşteptările reprezentan ilor comunită ilor epistemice, asocia ilor profesionale şi angajatori reprezentativi din domeniul aferent programului

Șabloanele de proiectare sunt folosite în toate sistemele sotware existente. Înțelegerea și utilizarea lor ajută la scrierea unui cod de calitate.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Insușirea principiilor teoretice prezentate la curs.	Examen final pe calculator	30%
10.5 Seminar/ laborator/ proiect	Capacitatea de a implementa algoritmii prezentați la curs în contextul unor probleme practice	Prezentarea unui proiect	40%
		Prezentarea programelor scrise de-a lungul semestrului	30%
10.6 Standard minim de performan ă			
Obținerea notei 5 la toate tipurile de evaluare (examen de laborator și examen scris)			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17.07.2025 și aprobată în ședința de Consiliu al facultății din data de 17.07.2025.

Conf. Dr. Ion Gabriel Stan Decan	Conf. Dr. Nicușor Minculete Director de departament

Lect. Dr. Ichim Vasile Sebastian Titular de curs	Lect. Dr. Ichim Vasile Sebastian Titular de laborator

Notă:

1) Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

2) Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

3) Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

4) Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii de licență[1)]	Informatică
1.5 Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	INFORMATICĂ APLICATĂ/ Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Procesarea imaginilor cu ajutorul algoritmilor inteligenți							
2.2 Titularul activităților de curs				Lect. dr. Plajer Ioana Cristina Conf. Dr. Alexandra Băicoianu				
2.3 Titularul activităților de laborator				Lect. dr. Plajer Ioana Cristina Drd. Cristian George Fieraru				
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar/ laborator/ proiect	0/28/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie și notițe	33
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	10
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri	40
Tutorat	7
Examinări	4
Alte activități.....	0

3.7 Total ore studiu individual	94
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Algoritmi, limbaj programare C++, C# sau Python cunoștințe de matematică de nivel de liceu - profil real, cunoștințe minime de engleză
4.2 de competențe	• Elaborarea unor programe în C++, C# sau Python.

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	• Tablă, Proiector
5.2 de desfășurare a seminarului/ laboratorului/ proiectului	• Calculatoare, medii de programare C# - Microsoft Visual Studio

6. Competențe specifice acumulate

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt **Cunoștințe** R.Î.1.1. Absolventul poate să scrie cod clar și bine structurat în limbaje de programare specifice. R.Î.1.2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale. **Aptitudini** R.Î.1.3. Absolventul poate să utilizeze biblioteci și framework-uri pentru îmbunătățirea performanțelor și funcționalitatea aplicațiilor software. R.Î.1.4. Absolventul poate să dezvolte aplicații complexe cu cerințele propuse de către utilizatori. **Responsabilitate și autonomie** R.Î.1.5. Absolventul este capabil să organizeze, planifice și supravegheze dezvoltarea aplicațiilor și a cadrelor necesare pentru a crea un produs software complet, de la etapele inițiale de planificare până la testarea finală. CP4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale **Cunoștințe** R.Î.4.1. Absolventul poate să verifice corespondența dintre capacitățile, corectitudinea și eficienta unui algoritm sau a unui sistem intenționat și anumite specificații formale. R.Î.4.2. Absolventul poate să utilizeze modele formale și teorii matematice pentru a rezolva probleme din diverse domenii ale informaticii. **Aptitudini** R.Î.4.3. Absolventul poate să utilizeze concepte și tehnici matematice pentru a analiza și proiecta algoritmi și structuri de date folosind tehnologii specifice. R.Î.4.4. Absolventul poate să utilizeze modele formale pentru a proiecta sisteme informatice sigure și fiabile. R.Î.4.5. Absolventul poate să utilizeze modele formale pentru a verifica corectitudinea programelor. **Responsabilitate și autonomie** R.Î.4.6. Studentul/absolventul construiește etic și responsabil soluții IT sigure și scalabile, colaborând cu specialiști din domenii conexe. R.Î.4.7. Studentul/absolventul dezvoltă un mediu colaborativ și își asumă responsabilitatea pentru succesul livrării proiectelor la timp și conform cerințelor. Studentul/absolventul organizează echipe tehnice și gestionează ciclul de viață al proiectelor software..

Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. R.Î.1.3. Absolventul are capacitatea de a acționa în mod responsabil și etic în activitățile didactice și de cercetare, demonstrând integritate și respect față de colegi, studenți și comunitatea academică. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite. CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. R.Î.2.3. Absolventul poate aplica tehnici de comunicare empatică, ascultând activ și răspunzând adecvat nevoilor și preocupărilor colegilor, contribuind astfel la crearea unui mediu de lucru armonios. R.Î.2.4. Absolventul poate adapta stilul de comunicare în funcție de contextul cultural și diversitatea grupului, asigurându-se că mesajul este transmis eficient și este bine înțeles de toți participanții. CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î.3.1. Absolventul va utiliza metode și tehnici eficiente de învățare pentru a-și îmbunătăți performanța academică. R.Î.3.2. Absolventul va utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis. R.Î.3.3. Absolventul poate evalua critic sursele de informație, identificând cele mai relevante și credibile materiale pentru a sprijini deciziile informate și fundamentate.

7. Obiectivele disciplinei (reiesind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	• Cunoașterea și înțelegerea unor elemente de bază din procesarea imaginilor digitale și adaptarea lor la probleme reale. • Dezvoltarea capacității studenților de a căuta documentație potrivită și de a aplica cunoștințele teoretice în practică
7.2 Obiectivele specifice	• Dezvoltarea capacității studenților de a aborda probleme de procesare de imagine, de a implementa algoritmi inteligenți pentru rezolvarea acestora. • Dezvoltarea abilității studenților de a interpreta și evalua rezultatele produse de propriile programe, de a adapta metodele generale la probleme concrete.

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
1. Introducere în procesarea de imagini, reprezentarea imaginilor, relații între pixeli. Spații de culoare. Modificări de bază ale contrastului și luminozității.	Expunere Dialog	2	

2. Algoritmi de bază 2.1 Histograma. Binarizare 2.2 Filtrare trece-jos și trece-sus. Elemente de bază și exemple.	Explicație Dezbatere Informare	4	
3. Introducere în morfologie matematică.		2	
4. Algoritmi de codificare și compresie.		4	
5. Algoritmi evolutivi în procesarea imaginilor.		4	
6. Analiza și procesarea imaginilor bazată pe rețele neuronale.		4	
7. Analiza imaginilor medicale.		4	
8. Procesarea imaginilor multi și hiper - spectrale.		4	

Bibliografie

1. Gonzalez R.C., Woods R.E., „Digital Image Processing" 4th Edition, Pearson 2018
2. Burger W., Burge M. - Digital Image Processing - 2nd edition, Springer 2016
3. Plajer I. C., Fieraru C.G, Băicoianu A., "Procesarea imaginilor digitale ", Curs pentru învățământul la distanță 2024, https://ebooks.unitbv.ro/produs/procesarea-imaginilor-digitale-curs-pentru-invatamant-la-distanta-anul-ii-sem-ii/
4. Cuevas E., Zaldivar D., Perez-Cisnerosm M., "Applications of Evolutionary Computation in Image Processing and Pattern Recognition", Springer, 2016
5. Kim-Hui Yap, Ling Guan, Stuart William Perry, Hau San Wong, „Adaptive Image Processing. A Computational Intelligence Perspective", Second Edition, 2018
6. Zhang C., Jiang W., Zhang Y., Wang W., Zhao Q., Wang C., „Transformer and CNN Hybrid Deep Neural Network for Semantic Segmentation of Very-High-Resolution Remote Sensing Imagery", IEEE Transactions on Geoscience and Remote Sensing, 2022
7. Alam M., Wang J.F., Guangpei C., Yunrong LV, Chen Y., „Convolutional Neural Network for the Semantic Segmentation of Remote Sensing Images", Mobile Networks and Applications, Springer Link 2021
8. Khan M.Z., Gajendran M.K, Lee Y., Khan M.A, „Deep Neural Architectures for Medical Image Semantic Segmentation: Review", IEEE Access, 2021
9. Guo Y., Liu Y., Georgiou T., Lew M.S., „A review of semantic segmentation using deep neural networks", International Journal of Multimedia Information Retrieval, 2017
10. Amigo, Jose Manuel. „Hyperspectral imaging". Vol. 32. Elsevier, 2019.
11. Zheng, Yufeng, Erik Blasch, and Zheng Liu. „Multispectral image fusion and colorization". Vol. 481. Bellingham, Washington: SPIE press, 2018.

8.2 Laborator	Metode de predare-învățare	Număr de ore	Observații
1. Instalarea și familiarizarea cu aplicația în care se va lucra la laborator	Aplicații / programare / interpretarea și evaluarea rezultatelor	2	
2. Implementarea algoritmilor de thresholding și filtrare.		2	
3. Alegerea unor proiecte în echipe și elaborarea proiectelor pe baza algoritmilor inteligenți discutați la curs. Temele urmează să fie lansate din		20	

domenii practice ca medicină sau procesarea imaginilor satelitare.			
4. Evaluare finală		4	

Bibliografie:

1. Gonzalez R.C., Woods R.E., „Digital Image Processing" 4th Edition, Pearson 2018
2. Burger W., Burge M. - Digital Image Processing - 2nd edition, Springer 2016
3. Plajer I. C., Fieraru C.G, Băicoianu A., "Procesarea imaginilor digitale ", Curs pentru învățământul la distanță 2024, https://ebooks.unitbv.ro/produs/procesarea-imaginilor-digitale-curs-pentru-invatamant-la-distanta-anul-ii-sem-ii/
4. Cuevas E., Zaldivar D., Perez-Cisnerosm M., "Applications of Evolutionary Computation in Image Processing and Pattern Recognition", Springer, 2016
5. Kim-Hui Yap, Ling Guan, Stuart William Perry, Hau San Wong, „Adaptive Image Processing. A Computational Intelligence Perspective", Second Edition, 2018
6. Zhang C., Jiang W., Zhang Y., Wang W., Zhao Q., Wang C., „Transformer and CNN Hybrid Deep Neural Network for Semantic Segmentation of Very-High-Resolution Remote Sensing Imagery", IEEE Transactions on Geoscience and Remote Sensing, 2022
7. Alam M., Wang J.F., Guangpei C., Yunrong LV, Chen Y., „Convolutional Neural Network for the Semantic Segmentation of Remote Sensing Images", Mobile Networks and Applications, Springer Link 2021
8. Khan M.Z., Gajendran M.K, Lee Y., Khan M.A, „Deep Neural Architectures for Medical Image Semantic Segmentation: Review", IEEE Access, 2021
9. Guo Y., Liu Y., Georgiou T., Lew M.S., „A review of semantic segmentation using deep neural networks", International Journal of Multimedia Information Retrieval, 2017
10. Amigo, Jose Manuel. „Hyperspectral imaging". Vol. 32. Elsevier, 2019.

Zheng, Yufeng, Erik Blasch, and Zheng Liu. „Multispectral image fusion and colorization". Vol. 481. Bellingham, Washington: SPIE press, 2018

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunităților epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

Fișa disciplinei a fost elaborată după consultări cu partenerii din mediul academic și economic ai Facultății.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Atingerea obiectivelor educaționale ale disciplinei	Examen scris	20 %
	Formarea competențelor vizate prin conținutul disciplinei		
10.5 Laborator	Prezentarea la termen a proiectelor în echipe.	Evaluarea cu punctaj a proiectelor.	80%
10.6 Standard minim de performanță			
Conținuturi:			

Pentru nota la laborator: • predarea la termen a proiectelor de laborator elaborate în echipe de câte 3. Pentru nota 5 la examenul scris • modele de culoare • thresholding și hitograme • filtrarea imaginilor. • compresia imaginilor. • Imagini multispectrale **Restricții**: Pentru promovarea examenului este necesară o notă de minim 5 la laborator și media dintre nota de la examenul scris și cea de la laborator să fie minim 5.

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. dr. Ion Gabriel Stan	Director de departament Conf. dr. Nicușor Minculete
Titular curs Lect. dr. Ioana Cristina Plajer Conf. dr. Alexandra Băicoianu	Titulari laborator Lect. dr. Ioana Cristina Plajer Drd. Cristian George Fieraru

Notă:

1) Domeniul de studii – se alege una din variantele: Licență / Masterat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor / programelor de studii universitare în vigoare).

2) Ciclul de studii – se alege una din variantele: Licență / Masterat.

3) Regimul disciplinei (conținut); se alege una din variantele: DF (disciplină fundamentală) / DD (disciplină din domeniu) / DS (disciplină de specialitate) / DC (disciplină complementară) - pentru nivelul de licență ; DAP (disciplină de aprofundare) / DSI (disciplină de sinteză) / DCA (disciplină de cunoaștere avansată) - pentru nivelul de masterat.

4) Regimul disciplinei (obligativitate) – se alege una din variantele: DI (disciplină impusă) / DO (disciplină opțională) / DFac (disciplină facultativă).

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice și studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Braşov
1.2 Facultatea	Matematică şi Informatică
1.3 Departamentul	Matematică şi Informatică
1.4Domeniul de studii de licență[1)]	Informatică
1.5Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	Informatică Aplicată / Licențiat în Informatică

2.Date despre disciplină

2.1 Denumirea disciplinei	Grafică pe calculator							
2.2 Titularul activităților de curs				**Prof. Dr. Deaconu Adrian**				
2.3 Titularii activităților de seminar/ laborator/ proiect				**Asist. Dr. Spridon Delia**				
2.4 Anul de studiu	1	2.5 Semestrul	2	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[4)]	DI

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 seminar/laborator/ proiect	**0/2/0**
3.4 Total ore din planul de învățământ	40	din care: 3.5 curs	20	3.6 seminar/laborator/ proiect	0/20/0

Distribuția fondului de timp	**Ore**
Studiul după manual, suport de curs, bibliografie şi notițe	26
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	26
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	38
Tutoriat	14
Examinări	6
Alte activități	
3.7 Total ore de activitate a studentului	110
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	-
4.2 de competențe	programare în C, pointeri

5. Condiții (acolo unde este cazul)

5.1 de desfăşurare a cursului	Sală de curs cu capacitate 75 de locuri, tablă, videoproiector, acces internet
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	Laborator – laborator de Informatică, calculatoare cu diverse medii de programare în care se pot redacta aplicații grafice, acces internet

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt CP4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale
Competențe transversale	CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoştințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională

7. Obiectivele disciplinei (reieşind din grila competențelor specifice acumulate)

7.1 Obiectivul general al disciplinei	Familiarizarea cu elemente specifice graficii 2D pe calculator

7.2 Obiectivele specifice	Studentul să poată scrie aplicații ce folosesc grafică

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
I. Transformări geometrice 2D. Coordonate omogene.		2	
II. Curbe 2D			
Reprezentarea grafică a unei curbe parametrice continue pe un interval		2	
Polinom de interpolare. Formula Lagrange. Formula Newton		2	
Curba Coons		1	
Curba Bezier		2	
Curba B-Spline		1	
III. Elemente de geometrie computațională			
Formula pentru testare cotire stânga/dreapta/coliniaritate		1	
Puncte separate de o dreaptă		0,25	
Testare intersecție a două segmente	Curs	0,25	
Testare poligon dacă este convex	interactiv	0,25	
Testare punct dacă este interior unui poligon convex	Prelegere	0,25	
Testare punct dacă este interior unui poligon oarecare	Dialog	1	
Triunghiularea unui poligon oarecare	Dezbatere	1	
Formula ariei unui poligon		1	
IV. Fractali			
Elemente de grafică Turtle		0,25	
Mulțimea Cantor (Georg Cantor)		0,25	
Curba și fulgul de nea al lui Koch		0,25	
Curba C a lui Lévy		0,25	
Curba dragon		0,25	
Fractal tip plantă		0,5	
Triunghiul, covorul lui Sierpinski		0,25	
Mulțimea Mandelbrot		1	

Bibliografie Hobart and William Smith Colleges, *Introduction to Computer Graphics*, Geneva, NY, 2023			
8.2 Laborator	Metode de predare-învățare	Număr de ore	Observații
I. Clasă care aplică și reține compunere de transformări 2D succesive în coordonate omogene II. Reprezentarea grafică de Curbe 2D (curbă parametrică continuă pe un interval, polinom de interpolare, curba Coons, curba Bezier, curba B-Spline) III. Bibliotecă de funcții de geometrie computațională: formula pentru testare cotire stânga/dreapta/coliniaritate, puncte separate de o dreaptă, testare intersecție a două segmente, testare poligon dacă este convex, testare punct dacă este interior unui poligon convex, testare punct dacă este interior unui poligon oarecare, triunghiularea unui poligon oarecare, formula ariei unui poligon IV. Reprezentare grafică de diverși fractali: mulțimea Cantor, curba și fulgul de nea al lui Koch, curba C a lui Lévy, curba dragon, fractal tip planta, triunghiul și covorul lui Sierpinski, mulțimea Mandelbrot	Problematizare Proiect Învățare prin probleme	2 8 5 3	
Bibliografie Hobart and William Smith Colleges, *Introduction to Computer Graphics*, Geneva, NY, 2023			

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, ale asociațiilor profesionale şi ale angajatorilor reprezentativi din domeniul aferent programului

Conținuturile sunt în concordanță cu problematica monografiilor apărute în ultimii ani în literatura de specialitate.

10.Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4. SI (curs)	examen scris		50%
10.5. TC / AA / ST / L / P	prezentarea programelor scrise de-a lungul semestrului		50%
10.6. Standard minim de performanță			
Obținerea notei 5 la toate tipurile de evaluare			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Conf. dr. Ion Gabriel Stan **Decan**	**Conf. dr. Nicușor Minculete** **Director de departament**

Prof. dr. Adrian Marius Deaconu Titular de curs	Asist. dr. Delia Spridon Titular de laborator

Notă:

[1)] Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

[2)] Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

[3)] Regimul disciplinei (conținut) - se alege una din variantele: DF (disciplină fundamentală)/ DD (disciplină din domeniu)/ DS (disciplină de specialitate)/ DC (disciplină complementară) - pentru nivelul de licență; DAP (disciplină de aprofundare)/ DSI (disciplină de sinteză)/ DCA (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

[4)] Regimul disciplinei (obligativitate) - se alege una din variantele: DI (disciplină obligatorie)/ DO (disciplină opțională)/ DFac (disciplină facultativă);

[5)] Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii de licență[1)]	Informatică
1.5 Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	Informatică Aplicată / Licențiat în informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Programare logică și funcțională							
2.2 Titularul activităților de curs				Lector dr. Adelina Manea				
2.3 Titularul activităților de seminar/ laborator/ proiect				Asist. dr. Maria Nuțu				
2.4 Anul de studiu	III	2.5 Semestrul	II	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[4)]	DI

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 seminar/ laborator/ proiect	**0/2/0**
3.4 Total ore din planul de învățământ	40	din care: 3.5 curs	20	3.6 seminar/ laborator/ proiect	0/20/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie şi notițe	45
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	25
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	30
Tutoriat	5
Examinări	5
Alte activități.....	

3.7 Total ore de activitate a studentului	110
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Algoritmică și structuri de date • Limbaje de programare
4.2 de competențe	• Deprinderea de utilizare a unui IDE • Cunoștințe de programare pentru implementarea soluțiilor de recursivitate • Cunoștințe de programare pentru implementarea soluțiilor de backtracking

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	• resurse fizice: laptop, videoproiector, tablă • resurse virtuale: infrastructură pentru internet, sisteme software pentru programare logică – Prolog și pentru programare funcțională – Lisp

5.2 de desfășurare a seminarului/ laboratorului/ proiectului	• resurse fizice: laptop, videoproiector, tablă • resurse virtuale: infrastructură pentru internet, sisteme software pentru programare logică – Prolog și pentru programare funcțională – Lisp

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențe profesionale	CP4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale R.Î. 4.1. Absolventul poate să utilizeze modele formale și teorii matematice pentru a rezolva probleme din diverse domenii ale informaticii. R.Î. 4.2. Absolventul poate sa utilizeze concepte și tehnici matematice pentru a analiza și proiecta algoritmi și structuri de date folosind tehnologii specifice. R.Î. 4.3. Absolventul poate să utilizeze modele formale pentru a proiecta sisteme informatice sigure și fiabile. R.Î. 4.4. Absolventul poate să utilizeze modele formale pentru a verifica corectitudinea programelor. CP1 Programarea în limbaje de nivel înalt R.Î. 1.1. Absolventul poate să scrie cod clar si bine structurat in limbaje de programare specifice. R.Î. 1.2. Absolventul poate să utilizeze biblioteci și framework-uri pentru îmbunătățirea performantelor si funcționalitatea aplicațiilor software. R.Î. 1.3. Absolventul poate să dezvolte aplicații complexe cu cerințele propuse de către utilizatori R.Î. 1.4. Absolventul poate să aplice metodele şi principiile folosind tehnologii specifice CP3 Utilizarea instrumentelor informatice în context interdisciplinar R.Î. 3.1. Absolventul poate utiliza instrumente informatice (folosind tehnologii specifice) pentru a colecta, analiza și interpreta date din diferite domenii de activitate R.Î. 3.2. Absolventul poate evalua și a selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare R.Î. 3.3. Absolventul poate să comunice și sa colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice.
Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională; R.Î. 1.1. Absolvenții vor fi capabili să de a-si organiza și gestiona timpul și resursele pentru a atinge obiectivele de învățare R.Î. 1.2. Absolvenții vor fi capabili să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î. 3.1. Absolvenții vor utiliza metode și tehnici eficiente de învățare pentru a-si îmbunătăți performanța academică. R.Î. 3.2 Absolvenții vor utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis.

7. Obiectivele disciplinei (reieşind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	• Folosirea eficientă a sistemelor hardware și software cu scopul prelucrării datelor prin cunoașterea, înțelegerea și aplicarea conceptelor

	și principiilor programării logice și ale programării funcționale
7.2 Obiectivele specifice	• Identificarea situațiilor problemă care se pot rezolva eficient prin aplicații logice și/sau funcționale • Cunoașterea, înțelegerea și aplicarea conceptelor și principiilor programării logice și ale programării funcționale pentru implementarea soluțiilor unor probleme specifice • Identificarea la nivelul limbajelor de programare a aspectelor logice și funcționale și valorificarea lor în dezvoltarea de proiecte

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
Programare Logică			
C1. Paradigme de programare. Programare imperativă. Programare declarativă. Recursivitate. Introducere în programarea logică.	Curs interactiv Prelegere Expunere Explicație Dezbatere	2	
C2. Programare logică în Prolog. Fapte și reguli în Prolog. Întrebări în Prolog. Modele de flux. Exemple introductive. Variabile în Prolog. Predicate cu aritate multiplă. Reguli de definire a potrivirilor. Operatori matematici în Prolog. Expresii de comparații.		2	
C3. Liste în Prolog. Liste liniare și liste eterogene. Manipularea listelor în Prolog.	Curs interactiv Prelegere Expunere Explicație Dezbatere	2	
C4. Backtracking în Prolog. Predicatul ! și predicatul fail. Recursivitate în coadă.		2	
C5. Aplicații Prolog în Inteligența artificială. Prelucrarea Limbajului Natural. Probleme de satisfacere a constrângerilor, etc.		2	
Programare funcțională			
C6. Introducere în programare funcțională. Principiile programării funcționale. Exemple.	Curs interactiv Prelegere Expunere Explicație Dezbatere	2	
C7. Limbajul de programare LISP. Elemente de bază. Liste. Tipuri de funcții în LISP. Reguli de evaluare. Exemple.		2	
C8. Metoda variabilei colectoare. Mecanisme funcționale: Funcțiile APPLY și FUNCALL. Funcții MAP. Exemple.		2	
C9. Matrice în LISP. Structura de Arbore în LISP. Construirea si traversarea unui arbore.		2	
C10. Aplicații LISP in Inteligența artificială. Probleme de satisfacere a constrângerilor (prezentare cod în paralel Prolog)		2	
Bibliografie			

1. Gabriela Czibula, H.F. Pop - Elemente Avansate de Programare în Lisp și Prolog. Aplicații in Inteligența Artificială, Editura Albastră, Cluj, 2012
2. Dorin Bocu - Fundamentele programării logice. Inițiere în prolog, editura Albastra, Cluj, 2018
3. Dorin Bocu - Programare logică și funcțională, editura Universității Transilvania din Brașov, 2019
4. Ulle Endriss - Lecture Notes: An introduction in Prolog., Institute for Logic, Language and Computation, Universitett van Amsterdam. PDF
5. Artificial Intelligence: Structures and Strategies for Complex Problem Solving (accesibil online: https://www.cs.unm.edu/~luger/ai-final/)

8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore	Observații
Prolog			
L1. Elemente Introductive Prolog. Liste Liniare Prolog	Exemplul Exercițiul Lucru individual	2	
L2. Liste eterogene Prolog		2	
L3. Backtracking în Prolog		2	
L4. Aplicații ale Inteligenței Artificiale implementate în Prolog		2	
L5. Examen parțial practic Prolog		2	
LISP			
L6-L7 Manipularea listelor în LISP	Exemplul Exercițiul Lucru individual	4	
L8. Arbori în LISP. Funcții MAP		2	
L9. Aplicații ale Inteligenței Artificiale implementate în LISP		2	
L5. Examen parțial practic LISP		2	

Bibliografie

1. Ulle Endriss - Lecture Notes: An introduction in Prolog., Institute for Logic, Language and Computation, Universitett van Amsterdam. PDF
2. Artificial Intelligence: Structures and Strategies for Complex Problem Solving (accesibil online: https://www.cs.unm.edu/~luger/ai-final/)

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, ale asociaților profesionale şi ale angajatorilor reprezentativi din domeniul aferent programului

Conținuturile sunt în concordanță cu problematica monografiilor apărute în ultimii ani în literatura de specialitate și sunt adaptate la necesitățile programului de studii informatică.

10. Evaluare

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	• atingerea obiectivelor educaționale ale disciplinei • acumularea competențelor vizate prin conținutul disciplinei	Evaluare finală prin probă scrisă	20%
10.5 Laborator		Evaluare formativă prin rapoarte tehnice, proiecte, aplicații software, activități participative	80%
10.6 Standard minim de performanță			
• prezentarea la termen a temelor incluse în evaluarea continuă			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025

Conf. dr. Ion-Gabriel STAN **Decan**	**Conf. dr. Nicușor MINCULETE** **Director de departament**
Lect. dr. Adelina - Loredana MANEA **Titular de curs**	**Asist. dr. Maria NUȚU** **Titular de laborator**

Notă:

[1)] Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

[2)] Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

[3)] Regimul disciplinei (conținut) - se alege una din variantele**: DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

[4)] Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

[5)] Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea *Transilvania* din Braşov
1.2 Facultatea	Matematică şi Informatică
1.3 Departamentul	Matematică şi Informatică
1.4 Domeniul de studii	Informatică Aplicată
1.5 Ciclul de studii[1)]	Licență
1.6 Programul de studii/ Calificarea	Informatică Aplicată / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei				**ELABORAREA LUCRĂRII DE LICENȚĂ**				
2.2 Titularul activităților de curs						Conf. Dr. Alexandra Băicoianu – coordonator program studii (și coordonatorii științifici ai proiectelor de licență)		
2.4 Anul de studiu	3	2.5 Semestrul	6	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[2)]	DS
							Obligativitate[3)]	DI

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

3.1 Număr de ore pe săptămână	6	din care: 3.2 curs		0	3.3 seminar/ laborator/ proiect	0/0/6
3.4 Total ore din planul de învățământ	60	din care: 3.5 curs		0	3.6 seminar/ laborator/ proiect	0/0/60

	Distribuția fondului de timp	ore
	Studiul după manual, suport de curs, bibliografie şi notițe	16
	Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	20
	Pregătire seminarii/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	40
	Tutoriat	14
	Examinări	-
	Alte activități.....	

3.7 Total ore studiu individual	90	
3.8 Total ore pe semestru	150	
3.9 Numărul de credite	*5*	

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Nu este cazul
4.2 de competențe	• Nu este cazul

5. Condiții (acolo unde este cazul)

5.1 de desfăşurare a cursului	Nu este cazul
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	• Nu este cazul

6. Competen e specifice acumulate

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt **Cunoștințe** R.Î.1.2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale. **Aptitudini** R.Î.1.3. Absolventul poate să utilizeze biblioteci și framework-uri pentru îmbunătățirea performanțelor și funcționalitatea aplicațiilor software. **Responsabilitate și autonomie** R.Î.1.5. Absolventul este capabil să organizeze, planifice și supravegheze dezvoltarea aplicațiilor și a cadrelor necesare pentru a crea un produs software complet, de la etapele inițiale de planificare până la testarea finală. CP2. Gestionarea ciclului de viață al aplicațiilor informatice **Cunoștințe** R.Î.2.1. Absolventul poate să parcurgă ciclul de viață al unui sistem informatic, începând cu ancheta preliminară și terminând cu exploatarea și menținerea sistemului. R.Î.2.2. Absolventul poate să identifice și să evalueze riscurile asociate cu fiecare fază a ciclului de viață. **Aptitudini** R.Î.2.6. Absolventul poate să aplice măsuri de securitate pentru protejarea datelor pe tot parcursul ciclului de viață. R.Î.2.7. Absolventul poate realiza documentație tehnică și manuale de utilizare pentru aplicațiile software, facilitând astfel utilizarea și întreținerea acestora de către specialiști din diverse domenii. CP3. Utilizarea instrumentelor informatice în context interdisciplinar **Cunoștințe** R.Î.3.2. Absolventul poate evalua și selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare, având abilitatea de a analiza nevoile specifice ale unui proiect și de a alege soluții tehnologice eficiente. **Aptitudini** R.Î.3.3. Absolventul poate implementa metode de testare și evaluare a software-ului, inclusiv testarea interactivă și feedback-ul utilizatorilor din diverse discipline, pentru a asigura calitatea și funcționalitatea aplicațiilor create. **Responsabilitate și autonomie** R.Î.3.5. Absolventul poate să comunice și să colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice, dezvoltând competențe de lucru în echipă și abilități de gestionare a proiectelor interdisciplinare. CP4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale **Cunoștințe** R.Î.4.1. Absolventul poate să verifice corespondența dintre capacitățile, corectitudinea și eficienta unui algoritm sau a unui sistem intenționat și anumite specificații formale. R.Î.4.2. Absolventul poate să utilizeze modele formale și teorii matematice pentru a rezolva probleme din diverse domenii ale informaticii. **Aptitudini** R.Î.4.3. Absolventul poate să utilizeze concepte și tehnici matematice pentru a analiza și proiecta algoritmi și structuri de date folosind tehnologii specifice. R.Î.4.4. Absolventul poate să utilizeze modele formale pentru a proiecta sisteme informatice sigure și fiabile. R.Î.4.5. Absolventul poate să utilizeze modele formale pentru a verifica corectitudinea programelor. **Responsabilitate și autonomie** R.Î.4.6. Studentul/absolventul construiește etic și responsabil soluții IT sigure și scalabile, colaborând cu specialiști din domenii conexe. CP5. Proiectarea, dezvoltarea şi gestiunea bazelor de date

	CP7. Analiza, proiectarea, dezvoltarea, administrarea și mentenanța sistemelor software pentru WEB şi Internet
Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoştințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională

7. Obiectivele disciplinei (reieşind din grila competen elor specifice acumulate)

7.1 Obiectivul general al disciplinei	Disciplina de Elaborarea lucrării de licență este concepută să ofere studenților de la programul de studii de Informatică aplicată materialul necesar dezvoltării de proiecte complexe. Se urmărește cunoașterea și aprofundarea tehnologiilor utile pentru rezolvarea unei probleme, justificarea alegerilor, implementarea unui prototip și documentarea lui ca lucrare de licență.
7.2 Obiectivele specifice	Disciplina se constituie ca o disciplină de specialitate a domeniului fiind cea care oferă absolvenților cunoștințe sintetice și analitice pentru rezolvarea problematicilor dedicate activităților practice.

8. Conținuturi

8.1 Curs	Metode de predare	Observații
Nu este cazul		
8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Observații
Nu este cazul		
Bibliografie: Selectată împreună cu cadrul didactic coordonator.		

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, asociațiilor profesionale şi angajatori reprezentativi din domeniul aferent programului

Coroborarea se aplică în *Acordurile de parteneriat* şi în *Conven iile de Practică* încheiate cu partenerii din mediul socio-economic ai Facultății/Universității.

10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
10.4 Curs	Aplicarea metodelor specifice de rezolvare pentru o temă dată Documentarea tehnicilor și tehnologiilor folosite, a alternativelor și justificarea alegerii Docuemntarea implementării: strcutura proiectului, arhitectură și modularizare, algoritmi, model de date, contribuții personale	Nu este cazul	0%
10.5 Seminar/ laborator/ proiect		Evaluare continuă 1) prezentarea lucrării în faze parțiale (vize) 2) prezentarea aplicației în fața comisiei 3) răspuns la întrebările comisiei legate de lucrare	100%
10.6 Standard minim de performan ă			
Stabilit de fiecare coordonator științific Prevederi stabilite în ghidul dedicat susținerii examenului de licență			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. Dr. Ion Gabriel Stan	Director de departament Conf. Dr. **Nicușor Minculete**
Titular de curs Conf. Dr. Alexandra Băicoianu	Titular de laborator Conf. Dr. Alexandra Băicoianu

Notă:

[4)] Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

[5)] Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

[6)] Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

[7)] Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

[8)] Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii de licență[1)]	Informatică
1.5 Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	INFORMATICĂ APLICATĂ / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Sisteme de Gestiune a Bazelor de Date							
2.2 Titularul activităților de curs				Conf. dr. Honorius Gâlmeanu				
2.3 Titularul activităților de seminar/ laborator/proiect				Conf. dr. Honorius Gâlmeanu				
2.4 Anul de studiu	3	2.5 Semestrul	2	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[4)]	DI

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

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	40	din care: 3.5 curs	20	3.6 seminar/ laborator/ proiect	0/20/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie și notițe	50
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	10
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri	40
Tutoriat	7
Examinări	3
Alte activități.....	0

3.7 Total ore de activitate a studentului	110
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Algoritmi și structuri de date
4.2 de competențe	• R.Î.1.1. Absolventul poate să scrie cod clar și bine structurat în limbaje de programare specifice. • R.Î.4.2. Absolventul poate să utilizeze modele formale și teorii matematice pentru a rezolva probleme din diverse domenii ale informaticii. • R.Î.4.3. Absolventul poate să utilizeze concepte și tehnici matematice pentru a analiza și proiecta algoritmi și structuri de date folosind tehnologii specifice.

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	• Tablă, Proiector

5.2 de desfășurare a seminarului/ laboratorului/ proiectului	• Calculatoare cu posibilitatea instalării MySQL server și MySQL Workbench, mașină virtuală Java (JVM) / Oracle VirtualBox instalat capabil să ruleze mașină virtuală Linux, editor de text, proiector, tablă

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt **Cunoștințe** R.Î.1.1. Absolventul poate să scrie cod clar și bine structurat în limbaje de programare specifice. R.Î.1.2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale. **Aptitudini** R.Î.1.4. Absolventul poate să dezvolte aplicații complexe cu cerințele propuse de către utilizatori. **Responsabilitate și autonomie** R.Î.1.5. Absolventul este capabil să organizeze, planifice și supravegheze dezvoltarea aplicațiilor și a cadrelor necesare pentru a crea un produs software complet, de la etapele inițiale de planificare până la testarea finală. CP5. Proiectarea, dezvoltarea şi gestiunea bazelor de date **Cunoștințe** R.Î.5.1. Absolventul poate să proiecteze scheme de baze de date eficiente și bine structurate. R.Î.5.2. Absolventul poate să asigure integritatea și securitatea datelor din bazele de date. **Aptitudini** R.Î.5.3. Absolventul poate să utilizeze limbaje de interogare pentru a accesa și manipula datele din bazele de date. R.Î.5.4. Absolventul poate să implementeze și să gestioneze sisteme de gestiune a bazelor de date (SGBD). R.Î.5.5. Absolventul poate să optimizeze performanța bazelor de date prin indexare și alte tehnici relevante. R.Î.5.6. Absolventul poate dezvolta interogări SQL complexe pentru a extrage, modifica și gestiona datele din baze de date, demonstrând abilități avansate de manipulare a datelor. R.Î.5.7. Absolventul aplică metode de migrare și de conversie a datelor existente pentru a transfera sau a converti date între formate, sisteme de stocare sau sisteme informatice. **Responsabilitate și autonomie** R.Î.5.8. Absolventul elaborează documentația care conține informații despre baza de date care este relevantă pentru utilizatorii finali. R.Î.5.9. Absolventul aplică politici de securitate informatică, implementând politici, metode și reglementari pentru securitatea datelor și informațiilor pentru a respecta principiile confidențialitatii, integrității și disponibilității.

Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. R.Î.1.3. Absolventul are capacitatea de a acționa în mod responsabil și etic în activitățile didactice și de cercetare, demonstrând integritate și respect față de colegi, studenți și comunitatea academică. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite. CT2 Desfăşurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. R.Î.2.3. Absolventul poate aplica tehnici de comunicare empatică, ascultând activ și răspunzând adecvat nevoilor și preocupărilor colegilor, contribuind astfel la crearea unui mediu de lucru armonios. R.Î.2.4. Absolventul poate adapta stilul de comunicare în funcție de contextul cultural și diversitatea grupului, asigurându-se că mesajul este transmis eficient și este bine înțeles de toți participanții. CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoştințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î.3.1. Absolventul va utiliza metode și tehnici eficiente de învățare pentru a-și îmbunătăți performanța academică. R.Î.3.2. Absolventul va utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis. R.Î.3.3. Absolventul poate evalua critic sursele de informație, identificând cele mai relevante și credibile materiale pentru a sprijini deciziile informate și fundamentate.

7. Obiectivele disciplinei (reieșind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	• Dezvoltarea de aplicații cu baze de date relaționale. Configurarea serverului MySQL. Proiectarea bazei de date având în vedere eficiența în operare. Realizarea de query-uri reapide.
7.2 Obiectivele specifice	• Proiectarea modelului de date folosind modelul relațional şi algebra relațională • Realizarea de interogări eficiente folosind limbajul SQL • Formularea modelului pornind de la descrierea problemei • Impunerea constrângerilor în modelul bazei de date • Alegerea informată a indecșilor și formularea tranzacțiilor. Alegerea nivelelor de izolare

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații

1. Proprietățile DBMS-urilor. Concepte cheie. Roluri. Modelul relațional și algebra relațională: select, project, join, union, difference, rename. Cross product. Natural join. Theta join.	Expunere Probleme rezolvate la curs Quiz-uri Discuție pe problemele rezolvate	2	
2. SQL: select, relații și operatori, subquery-uri în clauza WHERE. Subquery-uri în FROM și SELECT.		2	
3. SQL: operatori join, agregare, valori NULL, modificarea datelor		2	
4. Dependințe funcționale, forma BCNF		2	
5. Dependințe cu valori multiple și forma 4NF. Contraexemple.		2	
6. Descrierea modelelor relaționale cu UML		2	
7. Indecși. Tranzacții. Proprietățile tranzacțiilor. Nivele de izolare		2	
8. Constrângeri. Integritate referențială. Triggers.		2	
9. Triggers și proceduri stocate		2	
10. Rezolvare supervizată de probleme pentru examen	Dialog	2	
Bibliografie • H. Gâlmeanu, „Baze de date relaționale cu MySQL", Editura Universității Transilvania din Brașov, 2021 • R. Elmasri, S. Navathe, „Fundamentals of Database Systems (7th edition)", Pearson, 2017 • H. Garcia-Molina, J. Ullman, J. Widom, "Database Systems: The Complete Book", Pearson, 2nd ed, 2008			
8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore	Observații
1. Algebra relațională. Rezolvarea de exerciții cu interpretorul de algebră relațională.	Exerciții rezolvate. Exerciții de rezolvat. Verificarea temelor Evaluarea temelor	2	
2. Configurarea serverului MySQL. Crearea bazei de date. Query-uri SQL. Rezolvarea de exerciții cu MySQL workbench.		2	
3. Exerciții pe reprezentarea unui graf stocat în baza de date.		2	
4. Exerciții cu modificarea bazei de date.		2	
5. Proiectarea modelului relațional și a query-urilor necesare pornind de la enunțul problemei. Proiectarea bazei și a query-urilor. Comparația cu rezultatele deja calculate.		8	
6. Rezolvarea generică a problemei de generare a tuplelor pentru dependințele cu valori multiple.		2	
7. Abordarea unitară a unei aplicații SGBD folosind Object Relational Model în Python. Comparație de performanță între implementări.		2	
Bibliografie • https://www.mysql.com/products/workbench/ • https://www.virtualbox.org/ • https://ponyorm.com/ • https://developers.google.com/edu/python/			

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunităților epistemice, ale asociaților profesionale și ale angajatorilor reprezentativi din domeniul aferent programului

Fișa disciplinei a fost elaborată după consultări cu partenerii din mediul academic și economic ai Facultății.

10. Evaluare

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Atingerea obiectivelor	Examen scris tip test grilă	60%

	educaționale ale disciplinei Formarea competențelor vizate prin conținutul disciplinei		
10.5 Seminar/ laborator/ proiect	Implementarea aplicațiilor formulate, conform cu baremul pentru fiecare temă.	Evaluare cu notă a temelor de laborator	40%

10.6 Standard minim de performanță

Conținuturi:

Pentru nota la **laborator**:

- predarea la timp a temelor de laborator, bazate pe noțiunile predate la curs, și obținerea unei o medii de minim 5 în cadrul evaluării de laborator.

Pentru nota la **examenul scris**

- înțelegerea funcționării unui query SQL şi capacitatea de a construi un query minimal
- definirea corectă a noțiunilor specifice SGBD şi operarea cu ele

Restricții:

Pentru promovarea examenului sunt necesare o notă de minim 5 la laborator și o notă de minim 5 la examenul scris.

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. dr. Ion Gabriel Stan	Director de departament Conf. dr. Nicușor Minculete
Titular curs Conf. dr. Honorius Gâlmeanu	Titular laborator / seminar Conf. dr. Honorius Gâlmeanu

Notă:

1) Domeniul de studii – se alege una din variantele: Licență / Masterat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor / programelor de studii universitare în vigoare).

2) Ciclul de studii – se alege una din variantele: Licență / Masterat/ Doctorat.

3) Regimul disciplinei (conținut); se alege una din variantele: DF (disciplină fundamentală) / DD (disciplină din domeniu) / DS (disciplină de specialitate) / DC (disciplină complementară) - pentru nivelul de licență ; DAP (disciplină de aprofundare) / DSI (disciplină de sinteză) / DCA (disciplină de cunoaștere avansată) - pentru nivelul de masterat.

4) Regimul disciplinei (obligativitate) – se alege una din variantele: DI (disciplină impusă) / DO (disciplină opțională) / DFac (disciplină facultativă).

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice și studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea *Transilvania* din Braşov
1.2 Facultatea	Matematică şi Informatică
1.3 Departamentul	Matematică şi Informatică
1.4 Domeniul de studii	Informatică Aplicată
1.5 Ciclul de studii[1)]	Licență
1.6 Programul de studii/ Calificarea	Informatică Aplicată / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	**Antreprenoriat și strategii de afaceri**							
2.2 Titularul activităților de curs	Conf. Dr. Dana Adriana Lupşa-Tătaru							
2.3 Titularul activităților de seminar/ laborator/ proiect	Dr. Andreea Her anu							
2.4 Anul de studiu	3	2.5 Semestrul	6	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[2)]	DS
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs		2	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	40	din care: 3.5 curs		20	3.6 seminar/ laborator/ proiect	0/20/0

	Distribuția fondului de timp	ore
	Studiul după manual, suport de curs, bibliografie şi notițe	38
	Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	15
	Pregătire seminarii/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	45
	Tutoriat	6
	Examinări	6
	Alte activități.....	0

3.7 Total ore studiu individual	*110*	
3.8 Total ore pe semestru	*150*	
3.9 Numărul de credite	*5*	

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Management • Management strategic
4.2 de competențe	• Gândire analitică, sintetică

5. Condiții (acolo unde este cazul)

5.1 de desfăşurare a cursului	Sală de minim 80 de locuri, videoproiector
5.2 de desfăşurare a seminarului/ laboratorului/	• Instalat NotePad++ • Internet

proiectului	

6. Competen e specifice acumulate

Competențe profesionale	CP3. Utilizarea instrumentelor informatice în context interdisciplinar **Cunoștințe** R.Î.3.2. Absolventul poate evalua și selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare, având abilitatea de a analiza nevoile specifice ale unui proiect și de a alege soluții tehnologice eficiente. **Aptitudini** R.Î.3.4. Studentul/absolventul dezvoltă interesul pentru literatura de specialitate națională și internațională, participarea la cursuri specializate, conferințe, simpozioane, etc. pentru dezvoltarea profesională şi personală continuă și adaptarea eficientă la noile descoperiri ştiin ifice. **Responsabilitate și autonomie** R.Î.3.5. Absolventul poate să comunice și să colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice, dezvoltând competențe de lucru în echipă și abilități de gestionare a proiectelor interdisciplinare.
Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite. CT2 Desfăşurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.3. Absolventul poate aplica tehnici de comunicare empatică, ascultând activ și răspunzând adecvat nevoilor și preocupărilor colegilor, contribuind astfel la crearea unui mediu de lucru armonios. CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoştințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î.3.3. Absolventul poate evalua critic sursele de informație, identificând cele mai relevante și credibile materiale pentru a sprijini deciziile informate și fundamentate.

7. Obiectivele disciplinei (reieşind din grila competen elor specifice acumulate)

7.1 Obiectivul general al disciplinei	Cursul are ca scop principal însușirea de către studenți a conceptelor, abordărilor, metodelor și tehnicilor necesare pentru inițierea și implementarea inițiativelor antreprenoriale.
7.2 Obiectivele specifice	• acumularea de cunostințe de bază privind atat continutul cat si rolul a ceea ce se numeste antreprenoriat.

	• insusirea de metode si tehnici privind abordarea initiativelor antreprenoriale • dezvoltarea abilitatilor si deprinderilor pentru punerea in practica a ideilor antreprenoriale. • dezvoltarea interesului pentru promovarea initiativelor antreprenoriale din perspectiva politicilor publice; insusirea unui comportament progresist in ceea ce priveste aplicarea si implementarea conceptelor specifice. • dezvoltarea capacitatii de abordare si gandire in spirit managerial si antreprenorial.nsușirea de către studenți a conceptelor, abordărilor, metodelor și tehnicilor necesare pentru inițierea și implementarea inițiativelor antreprenoriale.

8. Conținuturi

8.1 Curs	Metode de predare	Observații
Mindset antreprenorial. Ideea și oportunitate	Prelegeri Expuneri Dialog Problematizare Studiu de caz Exemple	2 ore
Testarea pieței. Prototiuri și testarea produsului		2 ore
Branding și comunicare. Organizarea firmei		2 ore
Analiza mediului intern și extern. Strategia de marketing		2 ore
Analiza competiției. Buyer persona		2 ore
Planul de lansare și obiective SMART		2 ore
Strategii în afaceri. Diferențierea		2 ore
Branding și comunicare. Organizarea firmei		2 ore
Resursa umană. Recrutare, selecție și motivare		2 ore
Aspecte financiare		2 ore
		Total 20 de ore

Bibliografie:

- *Entrepreneurship* - Bygrave, W., Zacharakis, A., John Wiley & Sons Inc., N.J., 2008

- *Entrepreneurship* - Hisrich, R., Peters, M., Shepherd, D., McGraw-Hill Irwin, New York, 2013
- *Entrepreneurship – Successfully launching new ventures* - Barringer, B., Irelamd, R.D., Pearson Education Limited, 2012
- *Elemente Fundamentale* - Stancioiu, I., Militaru, Gh. Management. Ed. Teora, Bucuresti, 1999
- *Startup Ecosystems and Entrepreneurship: An Integrated Perspective* - Dee, N., et al., Palgrave Macmillan, 2022.
- *Entrepreneurship: Theory, Process, and Practice* - Kuratko, D. F., Cengage Learning, 2023.

8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Observații
Plan afacere – prezentare și mindset antreprenorial	Exerciții Dialog Lucru în echipă Problematizare Studiu individual	2 ore
Plan afacere – testarea pieței		2 ore
Plan afacere – branding și comunicare		4 ore
Plan afacere – mediul intern și extern		2 ore
Plan afacere – strategia de marketing		4 ore
Plan afacere – organizare și planificare		2 ore
Plan afacere – organizare și resurse umane		2 ore
Plan afacere – aspect financiare		2 ore
		Total 20 de ore

Bibliografie:

- *Entrepreneurship* - Bygrave, W., Zacharakis, A., John Wiley & Sons Inc., N.J., 2008
- *Entrepreneurship* - Hisrich, R., Peters, M., Shepherd, D., McGraw-Hill Irwin, New York, 2013
- *Entrepreneurship – Successfully launching new ventures* - Barringer, B., Irelamd, R.D., Pearson Education Limited, 2012
- *Elemente Fundamentale* - Stancioiu, I., Militaru, Gh. Management. Ed. Teora, Bucuresti, 1999
- *Startup Ecosystems and Entrepreneurship: An Integrated Perspective* - Dee, N., et al., Palgrave Macmillan, 2022.
- *Entrepreneurship: Theory, Process, and Practice* - Kuratko, D. F., Cengage Learning, 2023.

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, asociațiilor profesionale şi angajatori reprezentativi din domeniul aferent programului

Coroborarea se aplică în *Acordurile de parteneriat* şi în *Convențiile de Practică* încheiate cu partenerii din mediul socio-economic ai Facultății/Universității.

10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
10.4 Curs	Formarea competențelor vizate prin conținutul disciplinei. Atingerea obiectivelor educaționale din fișa disciplinei.	Evaluare finală **examen scris** care constă în rezolvarea unui subiect teoretic.	10%
10.5 Seminar/ laborator/ proiect		Evaluare continuă • **proiect** (P) care se prezintă la finalul semestrului și este evaluat cu notele P1 și P2 (P1: notă conținut proiect, P2: notă susținere proiect). • criteriu de promovare: nota P1 >= 5 & nota P2 >= 5	90%
10.6 Standard minim de performan ă			
Antreprenor vs. manager Strategii de promovare de produse/servicii online Elemente de organizare și planificare			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. Dr. Ion Gabriel Stan	Director de departament Conf. Dr. **Nicușor Minculete**
Titular de curs Conf. Dr. Dana Adriana Lupşa-Tătaru	Titular de laborator Dr. Andreea Her anu

Notă:

[1)] Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

[2)] Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

[3)] Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

[4)] Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

[5)] Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea *Transilvania* din Braşov
1.2 Facultatea	Matematică şi Informatică
1.3 Departamentul	Matematică şi Informatică
1.4 Domeniul de studii	Informatică Aplicată
1.5 Ciclul de studii[1)]	Licență
1.6 Programul de studii/ Calificarea	Informatică Aplicată / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei				**Full Stack development of web - based ML applications**				
2.2 Titularul activităților de curs					Conf. Dr. Alexandra Băicoianu			
2.3 Titularul activităților de seminar/ laborator/ proiect					Drd. Gyulai-Nagy Zoltán			
2.4 Anul de studiu	3	2.5 Semestrul	6	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[2)]	DS
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs		2	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	40	din care: 3.5 curs		20	3.6 seminar/ laborator/ proiect	0/20/0

	Distribuția fondului de timp	ore
	Studiul după manual, suport de curs, bibliografie şi note	38
	Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	15
	Pregătire seminarii/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	45
	Tutoriat	6
	Examinări	6
	Alte activități.....	0

3.7 Total ore studiu individual	*110*	
3.8 Total ore pe semestru	*150*	
3.9 Numărul de credite	*5*	

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Elemente de algoritmică și structuri de date (manipulare tipuri de date abstracte) • Noțiuni elementare din programare orientată pe obiecte în C++ (definirea unei clase, relații dintre clase, supraîncărcare de operatori, fluxuri in/out, etc)
4.2 de competențe	• Competențe informatice/digitale la nivel preuniversitar

5. Condiții (acolo unde este cazul)

5.1 de desfăşurare a cursului	Sală de minim 80 de locuri, videoproiector
5.2 de desfăşurare a seminarului/ laboratorului/	• Instalat Visual Studio Code, node.js (minim v16) si docker • Instalat NotePad++

proiectului	• Internet

6. Competen e specifice acumulate

Competențe profesionale	CP2. Gestionarea ciclului de viață al aplicațiilor informatice **Cunoștințe** R.Î.2.1. Absolventul poate să parcurgă ciclul de viață al unui sistem informatic, începând cu ancheta preliminară și terminând cu exploatarea și menținerea sistemului. **Aptitudini** R.Î.2.3. Absolventul poate să proiecteze soluții software eficiente și scalabile pentru fiecare fază a ciclului de viață. R.Î.2.5. Absolventul poate să monitorizeze și să optimizeze performanța sistemului în faza de exploatare. **Responsabilitate și autonomie** R.Î.2.8. Absolventul poate să lucreze în echipă pentru a parcurge ciclul de viață al unui sistem informatic. CP4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale **Cunoștințe** R.Î.4.1. Absolventul poate să verifice corespondența dintre capacitățile, corectitudinea și eficienta unui algoritm sau a unui sistem intenționat și anumite specificații formale. **Aptitudini** R.Î.4.4. Absolventul poate să utilizeze modele formale pentru a proiecta sisteme informatice sigure și fiabile. R.Î.4.5. Absolventul poate să utilizeze modele formale pentru a verifica corectitudinea programelor. **Responsabilitate și autonomie** R.Î.4.6. Studentul/absolventul construiește etic și responsabil soluții IT sigure și scalabile, colaborând cu specialiști din domenii conexe. CP7. Analiza, proiectarea, dezvoltarea, administrarea și mentenanța sistemelor software pentru WEB şi Internet **Cunoștințe** R.Î.7.1. Absolventul poate să identifice și să evalueze cerințelor sistemului software. R.Î.7.2. Absolventul poate să analizeze și să proiecteze arhitectura sistemului software. **Aptitudini** R.Î.7.3. Absolventul poate să identifice și să rezolve problemele de performanță și scalabilitate. R.Î.7.4. Absolventul poate să proiecteze interfețe WEB pentru utilizator, integrând baze de date și algoritmi eficienți pentru procesarea datelor. R.Î.7.5. Absolventul poate să administreze infrastructura hardware și software a sistemelor WEB și să monitorizeze performanța și securitatea sistemelor WEB. **Responsabilitate și autonomie** R.Î.7.8. Studentul/absolventul produce software și îl adaptează continuu la noile tehnologii și cerințe de piață.

Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. CT2 Desfăşurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoştințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î.3.1. Absolventul va utiliza metode și tehnici eficiente de învățare pentru a-și îmbunătăți performanța academică. R.Î.3.2. Absolventul va utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis.

7. Obiectivele disciplinei (reieşind din grila competen elor specifice acumulate)

7.1 Obiectivul general al disciplinei	Cursul se axează pe prezentarea metodologiilor de dezvoltare și mentenanță a aplicațiilor scrise în limbajul TypeScript, bazate pe extensii renumite care permit dezvoltarea modulelor ML. Folosind conceptele care vor fi introduse o să se poată scrie cod modern de TypeScript care respectă standardele arhitecturilor contemporane, utilizate în aplicații complexe și scalabile.
7.2 Obiectivele specifice	• aprofundare în utilizarea limbajului de programare TypeScript • introducere in utilizarea extensiilor React, Express, Docker si Tensorflow • dezvoltarea abilităților de proiectare al arhitecturilor de calitate • aplicarea unor șabloane/design patterns importante • introducere în ceea ce înseamnă lucrul în cloud, prin intermediul unor biblioteci potrivite și specifice pentru Google Cloud Platform. • dezvoltarea unei aplicații ce utilizează conceptele menționate anterior

8. Conținuturi

8.1 Curs	Metode de predare	Observații
Elemente introductive și o scurta descriere despre motivele istorice care au influențat dezvoltarea limbajului TypeScript. Definirea funcționalităților de baza în node.js: data types, modules, functions, garbage collector, promise, observable, rxjs,	Prelegeri	2 ore

package definitions, EcmaScript și diferențe între JavaScript și TypesScript.	Expuneri Dialog Problematizare Studiu de caz Exemple	
Programarea funcțională în React: inițializarea componentelor, hooks, Virtual DOM, contexts, Material UI, debugging		2 ore
Bazele aplicațiilor web: HTML, css, scss, animații, Tailwind, formulare, server side rendering, client rendering		2 ore
Crearea componentelor React și integrarea lor cu micro-servicii – Discuții și exemple		2 ore
Bazele micro-serviciilor node.js: REST API, JWT, OAUTH, SSL, Express si integrarea cu PostgreSQL și TypeORM		2 ore
Dezvoltarea aplicațiilor de tip REST API. Integrarea extensiei axios. Definirea rutelor într-o maniera MVC. Transmisia de date in format JSON si binar. Transmisia datelor în timp real prin websockets și prelucrarea acestora.		2 ore
Metode si abordări de desfășurare în condiții de producție. Analiza acestora din punct de vedere a complexității și a necesităților de mentenanță.		2 ore
Introducere în învățare automată. Prezentarea problemei de recunoaștere a imaginilor digitale ce reprezintă cifre scrise de mână. Analiza soluțiilor disponibile pentru implementarea cu Tensorflow.		2 ore
Desfășurarea aplicațiilor în Google Cloud Platform și funcționalitățile de procesare oferite de platformă. Introducere în metodologii de CI.		2 ore
Refactorizare a arhitecturii care permite conexiunea logică a capitolelor anterioare pentru finisarea aplicației. Evaluarea soluțiilor pentru eventuale probleme din medii reale de producție.		2 ore
		Total 20 de ore

Bibliografie:

- *Learning React: Modern Patterns for Developing React Apps* - Banks Alex
- *Learning TypeScript: Enhance Your Web Development Skills Using Type* -Safe JavaScript - Josh Goldberg
- *Full-Stack React, TypeScript, and Node* - David Choi
- *Arachitecting Modern Data Platforms* - Jan Kunigk
- *Practical Tensorflow.Js: Deep Learning in Web App Development* - Riviera Juan de Dios Santos
- *Full Stack Deep Learning* – Davidson, B. & Sirota, S.

8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Observații

Elemente introductive și o scurtă a limbajului TypeScript. Exerciții de aprofundare a funcționalităților de baza în node.js: data types, modules, functions, garbage collector, promise, observable, rxjs, package definitions, EcmaScript și diferențe între JavaScript si TypesScript.	Exerciții Dialog Lucru în echipă Problematizare Studiu individual	2 ore
Crearea unei aplicații în React ce utilizează: inițializarea componentelor, hooks, Virtual DOM, contexts, Material UI, debugging		2 ore
Extinderea aplicației React cu: HTML, css, scss, animații, Tailwind, fromulare, server side rendering, client rendering		2 ore
Crearea unui micro-serviciu node.js și exerciții de aprofundare pentru design-patterns din TypeScript.		2 ore
Integrarea în micro-serviciul anterior: Express, Axios, rute simple pentru transmisia datelor prin JSON, MVC.		2 ore
Crearea și rularea aplicațiilor în Docker și Google Cloud Platfrom. Aprofundarea funcționalităților din containere și imagini din mașini virtuale.		4 ore
Integrarea cu Tensorflow. Implementarea și antrenarea, conform exemplelor, al unui rețele neurale de recunoaștere a cifrelor scrise manual.		2 ore
Implementarea unui sistem de CI în Google Cloud Platfrom, pentru aplicațiile anterioare.		2 ore
Exerciții pentru „Best architectural approaches"		2 ore
		Total 20 de ore

Bibliografie:

- *Learning React: Modern Patterns for Developing React Apps* - Banks Alex
- *Learning TypeScript: Enhance Your Web Development Skills Using Type* -Safe JavaScript - Josh Goldberg
- *Full-Stack React, TypeScript, and Node* - David Choi
- *Arachitecting Modern Data Platforms* - Jan Kunigk
- *Practical Tensorflow.Js: Deep Learning in Web App Development* - Riviera Juan de Dios Santos
- *Full Stack Deep Learning* – Davidson, B. & Sirota, S.

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, asociațiilor profesionale şi angajatori reprezentativi din domeniul aferent programului

Coroborarea se aplică în *Acordurile de parteneriat* şi în *Conven iile de Practică* încheiate cu partenerii din mediul socio-economic ai Facultății/Universității.

10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
10.4 Curs		Evaluare finală **examen oral** care constă în	25%

	competențelor vizate prin conținutul disciplinei. Atingerea obiectivelor educaționale din fișa disciplinei.	rezolvarea și prezentarea orală unui subiect teoretic. Examenul final este eliminatoriu	
10.5 Seminar/ laborator/ proiect		Evaluare continuă • **proiect** (P) care se poate elabora în echipă de maxim 4 studenți. Proiectul se prezintă la finalul semestrului și este evaluat cu notele P1 și P2 (P1: notă conținut proiect, P2: notă susținere proiect). • criteriu de promovare: nota P1 >= 5 & nota P2 >= 5	75%
10.6 Standard minim de performan ă			
React: utilitate și beneficii Express: utilizare și beneficii Tensorflow: algoritmi Modele de arhitecturi scalabile: utilizare și beneficii			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. Dr. Ion Gabriel Stan	Director de departament Conf. Dr. **Nicușor Minculete**
Titular de curs Conf. Dr. Alexandra Băicoianu	Titular de laborator Drd. Gyulai-Nagy Zoltán

Notă:

[1)] Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

[2)] Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

[3)] Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

[4)] Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

[5)] Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii de Licență [1)]	Informatică
1.5 Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	Informatică Aplicată/Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Limbajul Natural în Machine Learning							
2.2 Titularul activităților de curs				Conf. dr. Honorius Gâlmeanu				
2.3 Titularul activităților de seminar/ laborator/ proiect				Conf. dr. Honorius Gâlmeanu				
2.4 Anul de studiu	3	2.5 Semestrul	6	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[4)]	DO

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 seminar/ laborator/ proiect	**0/2/0**
3.4 Total ore din planul de învățământ	40	din care: 3.5 curs	20	3.6 seminar/ laborator/ proiect	0/20/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie şi notițe	45
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	30
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	15
Tutoriat	14
Examinări	6
Alte activități.....	0

3.7 Total ore de activitate a studentului	110
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Cursuri universitare: Algebră liniară, Statistică matematică, Sisteme Computationale Inteligente / Inteligență Artificială
4.2 de competențe	• Programarea calculatoarelor, Limbajul Python

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	• Sală de curs cu tablă și videoproiector
5.2 de desfășurare a seminarului/ laboratorului/ proiectului	• Laborator de informatică cu stații de lucru / laptop

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențe profesionale	**CP1 Programarea în limbaje de nivel înalt** R.Î.1.1. Absolventul poate să scrie cod clar și bine structurat în limbaje de programare specifice. R.Î.1.2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale. **CP3 Utilizarea instrumentelor informatice în context interdisciplinar** R.Î.3.1. Absolventul poate utiliza instrumente informatice (folosind tehnologii specifice) pentru a colecta, analiza și interpreta date din diferite domenii de activitate. R.Î.3.2. Absolventul poate evalua și selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare, având abilitatea de a analiza nevoile specifice ale unui proiect și de a alege soluții tehnologice eficiente. R.Î.3.5. Absolventul poate să comunice și să colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice, dezvoltând competențe de lucru în echipă și abilități de gestionare a proiectelor interdisciplinare. **CP4 Utilizarea bazelor teoretice ale informaticii și a modelelor formale** R.Î.4.2. Absolventul poate să utilizeze modele formale și teorii matematice pentru a rezolva probleme din diverse domenii ale informaticii. R.Î.4.3. Absolventul poate să utilizeze concepte și tehnici matematice pentru a analiza și proiecta algoritmi și structuri de date folosind tehnologii specifice. R.Î.4.4. Absolventul poate să utilizeze modele formale pentru a proiecta sisteme informatice sigure și fiabile.
Competențe transversale	**CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională** R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. R.Î.1.3. Absolventul are capacitatea de a acționa în mod responsabil și etic în activitățile didactice și de cercetare, demonstrând integritate și respect față de colegi, studenți și comunitatea academică. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite. **CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoştințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională** R.Î.3.1. Absolventul va utiliza metode și tehnici eficiente de învățare pentru a-și îmbunătăți performanța academică. R.Î.3.2. Absolventul va utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis.

7. Obiectivele disciplinei (reieşind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	• Prezentarea conceptelor folosite pentru procesarea și generarea de text în limbaj natural
7.2 Obiectivele specifice	• Înțelegerea arhitecturii modelelor LLM și obținerea dexterității pentru codarea acestor modele. Înțelegerea performanțelor și limitărilor.

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
1. Introducere în modelele lingvistice. Aplicații ale LLM. Arhitectura tip transformer.	Curs interactiv	2	
2. Seturi de date mari. Arhitectura GPT.	Curs interactiv	2	
3. Tokens și embeddings. Crearea codificării contextualizate a cuvintelor. Word embeddings.	Curs interactiv	2	
4. Algoritmul Word2Vec. Contrastive Training. Encodarea pozițională a embedding-urilor.	Curs interactiv	2	
5. Rețele neuronale recurente. Modele de tip GRU și LSTM. Vanishing gradient. Aplicații pentru rețele neuronale recurente.	Curs interactiv	2	
6. Mecanismul attention. Modelarea secvențelor lungi.	Curs interactiv	2	
7. Implementarea self-attention. Causal attention mask. Attention heads.	Curs interactiv	2	
8. Codarea unei arhitecturi LLM. Layer normalization pentru normalizarea activărilor.	Curs interactiv	2	
9. Rețea feed-forward cu activări GELU. Shortcut connections. Modelul GPT. Generarea de text	Curs interactiv	2	
10. Preantrenarea. Antrenarea. Strategii pentru controlul antrenării aleatorii.	Curs interactiv	2	
Bibliografie 1. Jay Alammar, Maarten Grootendorst, „Hands-on Large Language Models", O'Reilly, 2024 2. Sebastian Raschka, „Build a Large Language Model from Scratch", Manning, 2024			
8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore	Observații
1. Tensori în PyTorch. Modelele computaționale și mecanismul autograd pentru antrenare. Modelul rețelei multistrat.	Seminar/Laborator	0/2	
2. Dataloaders. Bucla de antrenare. Controlul antrenării. Overfitting. Bias și variance.	Seminar/Laborator	0/2	
3. Crearea embedding-urilor. Implementarea algoritmului Word2Vec. Contrastive training.	Seminar/Laborator	0/2	
4. Învățarea cu attention.	Seminar/Laborator	0/2	
5. Învățarea cu self-attention.	Seminar/Laborator	0/2	
6. Crearea unui model LLM.	Seminar/Laborator	0/2	
7. Antrenarea unui mini-model LLM.	Seminar/Laborator	0/2	
8. Antrenarea unui mini-model LLM.	Seminar/Laborator	0/2	
9. Strategii pentru controlul antrenării.	Seminar/Laborator	0/2	
10. Strategii pentru controlul antrenării.	Seminar/Laborator	0/2	
Bibliografie 1. Jay Alammar, Maarten Grootendorst, „Hands-on Large Language Models", O'Reilly, 2024 2. Sebastian Raschka, „Build a Large Language Model from Scratch", Manning, 2024			

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, ale asociaților profesionale şi ale angajatorilor reprezentativi din domeniul aferent programului

Cursul oferă cunoștințe și deprinderi de bază de procesarea limbajului natural folosind rețele neuronale recurente și respectiv folosind mecanismul attention, caracteristice arhitecturii GPT.

10. Evaluare

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Acuratețea tratării subiectelor teoretice	Examen scris: subiecte de rezolvat tip grilă	60%
10.5 Seminar/ laborator/ proiect	Corectitudinea rezolvării problemelor și activitatea la curs şi seminar	Exerci ii de laborator de implementat în Python	40%
10.6 Standard minim de performanță			
Înțelegerea conceptelor de bază folosite la procesarea limbajului natural cu ajutorul rețelelor neuronale			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Conf. dr. Ion Gabriel Stan Decan	Conf. dr. Nicușor Minculete Director de departament
Titular de curs Conf. dr. Honorius GÂLMEANU	Titular de seminar/ laborator/ proiect Conf. dr. Honorius GÂLMEANU

Notă:

1) Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

2) Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

3) Regimul disciplinei (conținut) - se alege una din variantele**: DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

4) Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii de licență[1)]	Informatică
1.5 Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	INFORMATICĂ APLICATĂ/ Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Procesarea inteligentă a volumelor mari de informații vizuale							
2.2 Titularul activităților de curs				Conf. dr. Alexandra Băicoianu Lect. dr. Plajer Ioana Cristina				
2.3 Titularul activităților de laborator				Andrei Ghiurțu (Tremend, Publicis Sapient)				
2.4 Anul de studiu	3	2.5 Semestrul	5	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	40	din care: 3.5 curs	20	3.6 seminar/ laborator/ proiect	0/20/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie și notițe	43
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	15
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri	40
Tutorat	7
Examinări	5
Alte activități.....	0

3.7 Total ore studiu individual	110
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Algoritmi, limbaj programare C++, C# sau Python cunoștințe de matematică de nivel de liceu - profil real, cunoștințe minime de engleză
4.2 de competențe	• Elaborarea unor programe în C++, C# sau Python.

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	• Tablă, Proiector
5.2 de desfășurare a seminarului/ laboratorului/ proiectului	-

6. Competențe specifice acumulate

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt **Cunoștințe** R.Î.1.1. Absolventul poate să scrie cod clar și bine structurat în limbaje de programare specifice. R.Î.1.2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale. **Aptitudini** R.Î.1.3. Absolventul poate să utilizeze biblioteci și framework-uri pentru îmbunătățirea performanțelor și funcționalitatea aplicațiilor software. R.Î.1.4. Absolventul poate să dezvolte aplicații complexe cu cerințele propuse de către utilizatori. **Responsabilitate și autonomie** R.Î.1.5. Absolventul este capabil să organizeze, planifice și supravegheze dezvoltarea aplicațiilor și a cadrelor necesare pentru a crea un produs software complet, de la etapele inițiale de planificare până la testarea finală. CP4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale **Cunoștințe** R.Î.4.1. Absolventul poate să verifice corespondența dintre capacitățile, corectitudinea și eficienta unui algoritm sau a unui sistem intenționat și anumite specificații formale. R.Î.4.2. Absolventul poate să utilizeze modele formale și teorii matematice pentru a rezolva probleme din diverse domenii ale informaticii. **Aptitudini** R.Î.4.3. Absolventul poate să utilizeze concepte și tehnici matematice pentru a analiza și proiecta algoritmi și structuri de date folosind tehnologii specifice. R.Î.4.4. Absolventul poate să utilizeze modele formale pentru a proiecta sisteme informatice sigure și fiabile. R.Î.4.5. Absolventul poate să utilizeze modele formale pentru a verifica corectitudinea programelor. **Responsabilitate și autonomie** R.Î.4.6. Studentul/absolventul construiește etic și responsabil soluții IT sigure și scalabile, colaborând cu specialiști din domenii conexe. R.Î.4.7. Studentul/absolventul dezvoltă un mediu colaborativ și își asumă responsabilitatea pentru succesul livrării proiectelor la timp și conform cerințelor. Studentul/absolventul organizează echipe tehnice și gestionează ciclul de viață al proiectelor software..

Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. R.Î.1.3. Absolventul are capacitatea de a acționa în mod responsabil și etic în activitățile didactice și de cercetare, demonstrând integritate și respect față de colegi, studenți și comunitatea academică. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite. CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. R.Î.2.3. Absolventul poate aplica tehnici de comunicare empatică, ascultând activ și răspunzând adecvat nevoilor și preocupărilor colegilor, contribuind astfel la crearea unui mediu de lucru armonios. R.Î.2.4. Absolventul poate adapta stilul de comunicare în funcție de contextul cultural și diversitatea grupului, asigurându-se că mesajul este transmis eficient și este bine înțeles de toți participanții. CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î.3.1. Absolventul va utiliza metode și tehnici eficiente de învățare pentru a-și îmbunătăți performanța academică. R.Î.3.2. Absolventul va utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis. R.Î.3.3. Absolventul poate evalua critic sursele de informație, identificând cele mai relevante și credibile materiale pentru a sprijini deciziile informate și fundamentate.

7. Obiectivele disciplinei (reiesind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	• Dezvoltarea unei perspective interdisciplinare asupra procesării informațiilor vizuale la scară largă, integrând concepte din procesarea imaginilor, inteligență artificială și sisteme Big Data. • Familiarizarea cu tehnologii și metodologii moderne pentru analiza automată a imaginilor, incluzând algoritmi clasici și modele deep learning. • Înțelegerea provocărilor specifice prelucrării la scară mare: stocare, calcul distribuit, latență, volum și diversitate a datelor vizuale. • Formarea de competențe practice în utilizarea uneltelor software și a infrastructurilor cloud pentru procesarea inteligentă a imaginilor.
7.2 Obiectivele specifice	• Dezvoltarea capacității studenților de a înțelege și aplica conceptele fundamentale din procesarea imaginilor digitale, inclusiv tehnici de preprocesare, transformări spațiale și extracție de caracteristici.

	• Dezvoltarea abilității studenților de a implementa metode de procesare distribuită pentru volume mari de imagini, utilizând paradigme și unelte specifice Big Data. • Dezvoltarea capacității de a proiecta și automatiza fluxuri de prelucrare a imaginilor în infrastructuri cloud moderne, ținând cont de scalabilitate, eficiență și costuri. • Dezvoltarea abilității de a utiliza și adapta modele de deep learning pentru analiza imaginilor vizuale, cu aplicații în clasificare, segmentare și detecție de obiecte. • Dezvoltarea capacității studenților de a analiza și prelucra imagini provenite din surse complexe, precum cele multispectrale sau medicale, și de a selecta metode adecvate de interpretare. • Dezvoltarea abilității de a evalua performanța algoritmilor și sistemelor utilizate în procesarea informațiilor vizuale, din perspectiva acurateței, eficienței computaționale și robusteței.

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
1. Introducere în procesarea de imagini, reprezentarea imaginilor, preprocesare, transformări spațiale, instrumente și biblioteci specifice	Expunere Dialog Explicație Dezbatere Informare	2	
2. Noțiuni fundamentale de Big Data și Cloud Computing Paradigme Big data Stocare distribuită Introducere în infrastructura cloud (AWS/Azure)		2	
3. Înțelegerea informației vizuale prin Deep Learning Arhitecturi CNN Tehnici de augmentare, transfer learning Clasificare, segmentare semantică, detecție obiecte Integrarea modelelor deep learning în pipelines Big Data		4	
4. Procesarea și analiza imaginilor multispectrale și hiperspectrale Noțiuni de bază: domenii spectral, bandă, rezoluție Surse de date (ex. Sentinel, Landsat) Aplicabilitate (ex. agricultură, mediu, monitorizare urbană) Tehnici specifice (PCA, ICA, clasificare spectrală) Studii de caz și metode pentru interpretarea imaginilor din surse multispectrale.		4	
5. Procesarea imaginilor medicale la scară mare Formate specifice de imagini medicale Segmentarea automată de organe/structuri (MRI, CT) Deep learning pentru diagnostic (radiologie, patologie digitală) Provocări: anotare, confidențialitate, bias în seturi medicale		4	
6. Arhitecturi scalabile pentru procesarea informației vizuale		4	

Arhitecturi de referință pentru procesarea imaginilor în cloud (ex. serverless, microservicii) Evaluare performanța acestor arhitecturi și masurarea performanței în viața reală			

Bibliografie

1. Gonzalez R.C., Woods R.E., „Digital Image Processing" 4th Edition, Pearson 2018
2. Burger W., Burge M. - Digital Image Processing - 2nd edition, Springer 2016
3. Plajer I. C., Fieraru C.G, Băicoianu A., "Procesarea imaginilor digitale ", Curs pentru învățământul la distanță 2024, https://ebooks.unitbv.ro/produs/procesarea-imaginilor-digitale-curs-pentru-invatamant-la-distanta-anul-ii-sem-ii/
4. Elgendy, M. (2020). Deep Learning for Vision Systems. Manning Publications. ISBN: 9781617296192.
5. Russell, S., & Norvig, P. (2022). Artificial Intelligence: A Modern Approach (5th ed.). Pearson.
6. Rani, S., Bhambri, P., Kataria, A., Khang, A., & Sivaraman, A. K. (Eds.). (2023). Big Data, Cloud Computing and IoT: Tools and Applications. Chapman & Hall / Routledge.
7. Pani, S. K., Tripathy, S., Jandieri, G., Kundu, S., & Butt, T. A. (Eds.). (2021). Applications of Machine Learning in Big-Data Analytics and Cloud Computing. River Publishers.
8. Alam M., Wang J.F., Guangpei C., Yunrong LV, Chen Y., „Convolutional Neural Network for the Semantic Segmentation of Remote Sensing Images", Mobile Networks and Applications, Springer Link 2021

8.2 Laborator	Metode de predare-învățare	Număr de ore	Observații
1. Introducere în procesarea imaginilor digitale. Familiarizarea cu operații de bază asupra imaginilor.	Aplicații / programare / interpretarea și evaluarea rezultatelor	2	
2. Prelucrarea imaginilor în ecosisteme Big Data. Procesarea paralelă a unui volum mare de imagini și extragerea automată de metadate.		2	
3. Clasificarea imaginilor cu rețele neuronale convoluționale – Antrenarea și testarea unui model CNN pentru clasificarea automată a imaginilor.		4	
4. Analiza imaginilor multispectral		4	
5. Procesarea imaginilor medicale		4	
6. Proiect final: pipeline complet în cloud Dezvoltarea unui serviciu de procesare vizuală (ex. clasificare) și deploy într-un mediu cloud containerizat.		4	

Bibliografie:

1. Gonzalez R.C., Woods R.E., „Digital Image Processing" 4th Edition, Pearson 2018
2. Burger W., Burge M. - Digital Image Processing - 2nd edition, Springer 2016

3. Plajer I. C., Fieraru C.G, Băicoianu A., "Procesarea imaginilor digitale ", Curs pentru învățământul la distanță 2024, https://ebooks.unitbv.ro/produs/procesarea-imaginilor-digitale-curs-pentru-invatamant-la-distanta-anul-ii-sem-ii/
4. Alam M., Wang J.F., Guangpei C., Yunrong LV, Chen Y., „Convolutional Neural Network for the Semantic Segmentation of Remote Sensing Images", Mobile Networks and Applications, Springer Link 2021
5. Khan M.Z., Gajendran M.K, Lee Y., Khan M.A, „Deep Neural Architectures for Medical Image Semantic Segmentation: Review", IEEE Access, 2021
6. Guo Y., Liu Y., Georgiou T., Lew M.S., „A review of semantic segmentation using deep neural networks", International Journal of Multimedia Information Retrieval, 2017
7. Amigo, Jose Manuel. „Hyperspectral imaging". Vol. 32. Elsevier, 2019.
8. Elgendy, M. (2020). Deep Learning for Vision Systems. Manning Publications. ISBN: 9781617296192.
9. Russell, S., & Norvig, P. (2022). Artificial Intelligence: A Modern Approach (5th ed.). Pearson.
10. Rani, S., Bhambri, P., Kataria, A., Khang, A., & Sivaraman, A. K. (Eds.). (2023). Big Data, Cloud Computing and IoT: Tools and Applications. Chapman & Hall / Routledge.
11. Pani, S. K., Tripathy, S., Jandieri, G., Kundu, S., & Butt, T. A. (Eds.). (2021). Applications of Machine Learning in Big-Data Analytics and Cloud Computing. River Publishers.

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunităților epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

Fișa disciplinei a fost elaborată după consultări cu partenerii din mediul academic și economic ai Facultății.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Atingerea obiectivelor educaționale ale disciplinei	Examen scris	20 %
	Formarea competențelor vizate prin conținutul disciplinei		
10.5 Laborator	Prezentarea la termen a proiectului final.	Evaluarea cu punctaj a proiectelor.	80%
10.6 Standard minim de performanță			

Conținuturi:
Înțelegerea corectă a conceptelor fundamentale din procesarea imaginilor digitale.
Capacitatea de a prelucra un set de imagini la scară mică și medie, folosind instrumente software uzuale.
Cunoașterea principiilor de bază ale procesării distribuite și a rolului infrastructurilor de tip Big Data și Cloud în analiza datelor vizuale.
Abilitatea de a folosi o rețea neuronală pre-antrenată pentru o sarcină simplă de clasificare a imaginilor.
Restricții:
Pentru promovarea examenului este necesară o notă de minim 5 la proiectul final.

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. dr. Ion Gabriel Stan	Director de departament Conf. dr. Nicușor Minculete
Titular curs Conf. dr. Alexandra Băicoianu Lect. dr. Ioana Cristina Plajer	Titulari laborator Andrei Ghiurțu

Notă:

1) Domeniul de studii – se alege una din variantele: Licență / Masterat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor / programelor de studii universitare în vigoare).

2) Ciclul de studii – se alege una din variantele: Licență / Masterat.

3) Regimul disciplinei (conținut); se alege una din variantele: DF (disciplină fundamentală) / DD (disciplină din domeniu) / DS (disciplină de specialitate) / DC (disciplină complementară) - pentru nivelul de licență ; DAP (disciplină de aprofundare) / DSI (disciplină de sinteză) / DCA (disciplină de cunoaștere avansată) - pentru nivelul de masterat.

4) Regimul disciplinei (obligativitate) – se alege una din variantele: DI (disciplină impusă) / DO (disciplină opțională) / DFac (disciplină facultativă).

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice și studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea *Transilvania* din Braşov
1.2 Facultatea	Facultatea de Matematică şi Informatică
1.3 Departamentul	Departamentul de Matematică şi Informatică
1.4 Domeniul de studii de licență	Informatică
1.5 Ciclul de studii[1)]	Licență
1.6 Programul de studii/ Calificarea	INFORMATICĂ APLICATĂ / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	**PROGRAMARE PARALELĂ**							
2.2 Titularul activităților de curs				Prof. Dr. Sabin Tăbîrcă				
2.3 Titularul activităților de seminar/ laborator/ proiect				Drd. Bogdan Mihai Petre				
2.4 Anul de studiu	3	2.5 Semestrul	2	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[2)]	DS
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 laborator	**0 / 2 / 0**
3.4 Total ore din planul de învățământ	40	din care: 3.5 curs	20	3.6 seminar/ laborator/ proiect	0 / 20 / 0

Distribuția fondului de timp	**Ore**
Studiul după manual, suport de curs, bibliografie şi notițe	31
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	31
Pregătire seminarii/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	32
Tutoriat	10
Examinări	6
Alte activități.....	

3.7 Total ore studiu individual	110
3.8 Total ore pe semestru	150
3.9 Numărul de credite[4)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Programare in limbajul C
4.2 de competențe	• Competențe de programare la nivel universitar

5. Condiții (acolo unde este cazul)

5.1 de desfăşurare a cursului	Sală de curs cu capacitate 100 locuri, tablă, videoproiector
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	Seminar: sală de seminar, tablă; Laborator: laborator de Informatică, acces internet, produse software

6. Competențe specifice acumulate

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt. CP2. Gestionarea ciclului de viață al aplicațiilor informatice. CP3. Utilizarea instrumentelor informatice în context interdisciplinar.

Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională. CT2 Desfăşurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse

7. Obiectivele disciplinei (reieşind din grila competențelor specifice acumulate)

7.1 Obiectivul general al disciplinei	• Însușirea programării paralele in GPU.

7.2 Obiectivele specifice	• Cunoașterea principalelor componente ale limbajelor de programare pentru arhitecturi distribuite GPU: 1. Arhitectura NVIDIA pentru GPU. 2. Programare CUDA pentru sisteme GPU. • Algoritmi fundamentali paraleli. 1. Algoritmi GPU pentru sortari. 2. Algoritmi GPU pentru lucru cu matrici. 3. Algoritmi GPU pentru fractali. 4. Algoritmi GPU pentru AI.

8. Conținuturi

8.1 Curs	*Număr de ore*	*Metode de predare*	*Observații*
Arhitecturi NVIDIA Structura de memorii globale, shared, locale Structura computaționala CUDA	*6*		
Elemente de baza in programarea CUDA Algoritmi GPU pentru vectori. Algoritmi GPU pentru matrici. Algoritmi GPU pentru fractali. Algoritmi GPU de reducere.	*10*	Curs interactiv Prelegere Dialog	
Aplicatii ale programarii GPU Aplicatii GPU pentru AI. Aplicatii GPU pentru Bioinformatica.	*4*	Dezbatere	

Bibliografie

- Barry Wilkinson and Michael Allen (2005) Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers, Prentice-Hall.
- LLNL (Lawrence Livermore National Lab) Tutorial on Parallel Computing
- CUDA Official Page.

8.2 Seminar/ laborator/ proiect	*Număr de ore*	*Metode de predare-învățare*	*Observații*
Laborator			

Programare GPU pentru probleme cu numere - Procesare numere prime *Algoritmi GPU fundamentali* - Sortari simple externe si interne - Operati cu matrici e.g. inmultire, LU, inversare - Operatii de reducere - Fractali *Aplicatii GPU in* - Bioinformatica - AI	4 4 2 2 2 2 *2* *2*	 Lucru în grup Problematizare Proiect	
	2		

Bibliografie

- LLNL (Lawrence Livermore National Lab) Tutorial on Parallel Computing
- CUDA Official Page.

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, asociațiilor profesionale şi angajatori reprezentativi din domeniul aferent programului

10. Evaluare

Tip activitate	*10.1 Criterii de evaluare*	*10.2 Metode de evaluare*	*10.3 Pondere din nota finală*
10.4 Curs	• Atingerea obiectivelor educaționale ale disciplinei • Acumularea competențelor vizate prin conținutul disciplinei	Evaluare finală prin probă scrisă	50%
10.5 Seminar/ laborator/ proiect		Evaluare formativă prin rapoarte tehnice, proiecte, activități participative	50%
10.6 Standard minim de performanță			
• prezentarea la termene a temelor incluse în evaluarea continuă • programarea interfețelor si a bazelor de date pentru aplicații mobile			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17 iulie 2025 și aprobată în ședința de Consiliu al facultății din data de 17 iulie 2025.

Decan Conf. dr. Ion Gabriel Stan	Director de departament Conf. dr. Nicușor Miculete
Titular de curs Prof. dr. Sabin Tăbîrcă	Titular de laborator Drd. Bogdan Mihai Petre

Notă:

[1)] Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

[2)] Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

[3)] Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

[4)] Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

[5)] Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Institu ia de învă ământ superior	Universitatea „Transilvania” din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii	Informatică
1.5 Ciclul de studii[1)]	Licență
1.6 Programul de studii/ Calificarea	INFORMATICĂ APLICATĂ / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	Arhitectura aplicațiilor front-end folosind Angular							
2.2 Titularul activită ilor de curs				Lect. Dr. Ichim Vasile Sebastian				
2.3 Titularul activită ilor de seminar/ laborator/ proiect				Lect. Dr. Ichim Vasile Sebastian				
2.4 Anul de studiu	3	2.5 Semestrul	6	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Con inut[2)]	DS
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 seminar/ laborator/ proiect	**2**
3.4 Total ore din planul de învă ământ	40	din care: 3.5 curs	20	3.6 seminar/ laborator/ proiect	20

Distribu ia fondului de timp	ore
Studiul după manual, suport de curs, bibliografie şi noti e	45
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	20
Pregătire seminarii/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	20
Tutoriat	10
Examinări	10
Alte activită i	5
3.7 Total ore studiu individual	110
3.8 Total ore pe semestru	150
3.9 Numărul de credite[4)]	5

4. Precondi ii (acolo unde este cazul)

4.1 de curriculum	• Cursuri: Tehnici Web
4.2 de competen e	• Cunoștințe de programare

5. Condi ii (acolo unde este cazul)

5.1 de desfăşurare a cursului	• Sală de curs dotată cu videoproiector
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	• Laborator dotat cu calculatoare

6. Competen e specifice acumulate

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt Cunoștințe R.Î.1.1. Absolventul poate să scrie cod clar și bine structurat în limbaje de programare specifice. R.Î.1.2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale. Aptitudini R.Î.1.3. Absolventul poate să utilizeze biblioteci și framework-uri pentru îmbunătățirea performanțelor și funcționalitatea aplicațiilor software. R.Î.1.4. Absolventul poate să dezvolte aplicații complexe cu cerințele propuse de către utilizatori. Responsabilitate și autonomie R.Î.1.5. Absolventul este capabil să organizeze, planifice și supravegheze dezvoltarea aplicațiilor și a cadrelor necesare pentru a crea un produs software complet, de la etapele inițiale de planificare până la testarea finală. CP2. Gestionarea ciclului de viață al aplicațiilor informatice Cunoștințe R.Î.2.1. Absolventul poate să parcurgă ciclul de viață al unui sistem informatic, începând cu ancheta preliminară și terminând cu exploatarea și menținerea sistemului. Aptitudini R.Î.2.3. Absolventul poate să proiecteze soluții software eficiente și scalabile pentru fiecare fază a ciclului de viață. R.Î.2.7. Absolventul poate realiza documentație tehnică și manuale de utilizare pentru aplicațiile software, facilitând astfel utilizarea și întreținerea acestora de către specialiști din diverse domenii. Responsabilitate și autonomie R.Î.2.8. Absolventul poate să lucreze în echipă pentru a parcurge ciclul de viață al unui sistem informatic. CP3. Utilizarea instrumentelor informatice în context interdisciplinar Cunoștințe R.Î.3.2. Absolventul poate evalua și selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare, având abilitatea de a analiza nevoile specifice ale unui proiect și de a alege soluții tehnologice eficiente. Aptitudini R.Î.3.3. Absolventul poate implementa metode de testare și evaluare a software-ului, inclusiv testarea interactivă și feedback-ul utilizatorilor din diverse discipline, pentru a asigura calitatea și funcționalitatea aplicațiilor create. Responsabilitate și autonomie R.Î.3.5. Absolventul poate să comunice și să colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice, dezvoltând competențe de lucru în echipă și abilități de gestionare a proiectelor interdisciplinare. CP7. Analiza, proiectarea, dezvoltarea, administrarea și mentenanța sistemelor software pentru WEB şi Internet Cunoștințe R.Î.7.1. Absolventul poate să identifice și să evalueze cerințelor sistemului software. R.Î.7.2. Absolventul poate să analizeze și să proiecteze arhitectura sistemului software. Aptitudini R.Î.7.3. Absolventul poate să identifice și să rezolve problemele de performanță și scalabilitate. R.Î.7.4. Absolventul poate să proiecteze interfețe WEB pentru utilizator, integrând baze de date și algoritmi eficienți pentru procesarea datelor. R.Î.7.5. Absolventul poate să administreze infrastructura hardware și software a sistemelor WEB și să monitorizeze performanța și securitatea sistemelor WEB.

	R.Î.7.6. Absolventul poate să remedieze erorilor și vulnerabilitățile de securitate. R.Î.7.7. Absolventul poate să asigure compatibilitatea cu noile versiuni de tehnologii WEB. Responsabilitate și autonomie R.Î.7.8. Studentul/absolventul produce software și îl adaptează continuu la noile tehnologii și cerințe de piață.
Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. R.Î.1.3. Absolventul are capacitatea de a acționa în mod responsabil și etic în activitățile didactice și de cercetare, demonstrând integritate și respect față de colegi, studenți și comunitatea academică. R.Î.1.4. Absolventul are abilitatea de a-și organiza și prioritiza activitățile pentru a asigura o muncă eficientă și productivă, respectând termenele limită și obiectivele stabilite CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare interpersonală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. R.Î.2.3. Absolventul poate aplica tehnici de comunicare empatică, ascultând activ și răspunzând adecvat nevoilor și preocupărilor colegilor, contribuind astfel la crearea unui mediu de lucru armonios. R.Î.2.4. Absolventul poate adapta stilul de comunicare în funcție de contextul cultural și diversitatea grupului, asigurându-se că mesajul este transmis eficient și este bine înțeles de toți participanții. CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î.3.1. Absolventul va utiliza metode și tehnici eficiente de învățare pentru a-și îmbunătăți performanța academică. R.Î.3.2. Absolventul va utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis. R.Î.3.3. Absolventul poate evalua critic sursele de informație, identificând cele mai relevante și credibile materiale pentru a sprijini deciziile informate și fundamentate.

7. Obiectivele disciplinei (reieşind din grila competen elor specifice acumulate)

7.1 Obiectivul general al disciplinei	Dezvoltarea de aplicații front-end moderne utilizând Angular, o platformă puternică și popular dezvoltat de Google
7.1 Obiectivele specifice	• Înțelegerea principiilor fundamentale ale dezvoltării front-end moderne • Stăpânirea platformei Angular și a ecosistemului său • Dezvoltarea de aplicații web interactive, scalabile și ușor de întreținut • Aplicarea celor mai bune practici în dezvoltarea de software

8. Con inuturi

8.1 Curs	Metode de predare	Număr de ore
Introducere în dezvoltarea web, Angular în contextul actual	Expunere și discuție liberă	2
Componente și module	Expunere și discuție liberă	2
Servicii și comunicare	Expunere și discuție liberă	2
Directive și pipe-uri	Expunere și discuție liberă	4
Routing, guards și navigare	Expunere și discuție liberă	4
Programare reactivă folosind RxJs	Expunere și discuție liberă	4
Unit testing în Angular, Best practices	Expunere și discuție liberă	2

Bibliografie

1. Shyam Seshadri, Angular: Up and Running: Learning Angular, Step by Step, Editura O'Reilly, 2018
2. Aristeidis Bampakos, Pablo Deeleman, Learning Angular: A no-nonsense guide to building web applications with Angular 15, ediția a IV, Editura Packt, 2023
3. Doguhan Uluca, Angular for Enterprise Applications: Build scalable Angular apps using the minimalist Router-first architecture, Editura Packt, 2024

8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore
Manipularea obiectelor	Învățare prin probleme	2
Introducere în dezvoltarea web, Angular în contextul actual	Învățare prin probleme	2
Componente și module	Învățare prin probleme	2
Servicii și comunicare	Învățare prin probleme	2
Directive și pipe-uri	Învățare prin probleme	4
Routing, guards și navigare	Învățare prin probleme	4
Programare reactivă folosind RxJs	Învățare prin probleme	4

Bibliografie

1. Shyam Seshadri, Angular: Up and Running: Learning Angular, Step by Step, Editura O'Reilly, 2018
2. Aristeidis Bampakos, Pablo Deeleman, Learning Angular: A no-nonsense guide to building web applications with Angular 15, ediția a IV, Editura Packt, 2023
3. Doguhan Uluca, Angular for Enterprise Applications: Build scalable Angular apps using the minimalist Router-first architecture, Editura Packt, 2024

9. Coroborarea con inuturilor disciplinei cu aşteptările reprezentan ilor comunită ilor epistemice, asociațiilor profesionale şi angajatori reprezentativi din domeniul aferent programului

Dobândirea unei competențe solide într-un Framework UI precum Angular nu doar că facilitează accesul la oportunități de angajare, în special într-o piață unde Angular este extrem de căutat, dar oferă și un avantaj semnificativ pentru cei care doresc să dezvolte propriile proiecte, aplicații comerciale sau startup-uri.
Stăpânirea unui astfel de instrument permite o scalare eficientă a aplicațiilor și o înțelegere aprofundată a principiilor moderne de arhitectură software.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Însușirea principiilor teoretice prezentate la curs.	Examen final pe calculator	30%
10.5 Seminar/ laborator/ proiect		Prezentarea unui proiect	40%

	Capacitatea de a implementa algoritmii prezentați la curs în contextul unor probleme practice	Prezentarea programelor scrise de-a lungul semestrului	30%
10.6 Standard minim de performan ă			
Obținerea notei 5 la toate tipurile de evaluare (examen de laborator și examen scris)			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17.07.2025 și aprobată în ședința de Consiliu al facultății din data de 17.07.2025.

Conf. Dr. Ion Gabriel Stan Decan	Conf. Dr. Nicușor Minculete Director de departament
Lect. Dr. Ichim Vasile Sebastian Titular de curs	Lect. Dr. Ichim Vasile Sebastian Titular de laborator

Notă:

1) Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

2) Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

3) Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

4) Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Matematică și Informatică
1.4 Domeniul de studii de licență	Informatică
1.5 Ciclul de studii[2)]	Licență
1.6 Programul de studii/ Calificarea	Informatică Aplicată/Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	**Dezvoltarea aplicațiilor Web**							
2.2 Titularul activităților de curs				Conf. dr. Alexandra Băicoianu Gabriel Alexandru (Cornerstone)				
2.3 Titularul activităților de seminar/ laborator/ proiect				Gabriel Alexandru (Cornerstone)				
2.4 Anul de studiu	3	2.5 Semestrul	2	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[3)]	DS
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	**4**	din care: 3.2 curs	**2**	3.3 seminar/ laborator/ proiect	**0/2/0**
3.4 Total ore din planul de învățământ	40	din care: 3.5 curs	20	3.6 seminar/ laborator/ proiect	0/20/0

Distribuția fondului de timp	ore
Studiul după manual, suport de curs, bibliografie şi notițe	45
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	25
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	20
Tutoriat	10
Examinări	10
Alte activități	0

3.7 Total ore de activitate a studentului	110
3.8 Total ore pe semestru	150
3.9 Numărul de credite[5)]	5

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Algoritmi, limbaj programare C++, C#, sau Java, cunoștințe de POO, cunoștințe de matematică de nivel de liceu - profil real, cunoștințe minime de engleză
4.2 de competențe	Dezvoltare software într-unul din limbajele Java, Kotlin, C# sau TypeScript.

5. Condiții (acolo unde este cazul)

5.1 de desfăşurare a cursului	Tablă, Proiector, Internet

5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	Calculatoare, Tablă, Proiector, Internet

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt CP2. Gestionarea ciclului de viață al aplicațiilor informatice CP3. Utilizarea instrumentelor informatice în context interdisciplinar CP4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale
Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională. CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar și dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare și colaborare cu grupuri diverse CT3 Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare în limba română și într-o limbă de circulație internațională

7. Obiectivele disciplinei (reieşind din competențele specifice acumulate)

7.1. Obiectivul general al disciplinei	- Dezvoltarea unui produs software utilizând tehnologii, concepte și principii utilizate în industrie.
7.2. Obiectivele specifice	- Dezvoltarea capacității studenților de lucru în echipa pentru îndeplinirea unui scop comun; - Dezvoltarea abilității studenților de a pune în practică cunoștințele acumulate sub forma unui produs propriu.

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
Modul 1: Introducere și organizare Prima interacțiune cu studenții Prezentarea structurii cursului și a modulelor Sesiune de întrebări și răspunsuri	Expunere Dialog Explicație Dezbatere Studiu de caz	2	
Modul 2: Fundamentele aplicațiilor web Arhitectura client-server și tehnologii web moderne Securitate web de bază Instrumente de colaborare și versionare		2	
Modul 3: Dezvoltare specializată Backend Development Frontend Development (Optional) Mobile Development		4	
Modul 4: Performanță și optimizare Optimizarea aplicațiilor frontend și backend Lazy loading, caching, paginare, compresie Măsurarea performanței și profiling		4	

Modul 5: Testare și mentenanță Testare unitară, de integrare și end-to-end Frameworkuri de testare Debugging, logging și mentenanța aplicațiilor în producție		4	
Modul 6: Producție și bune practici Tool-uri utile în dezvoltarea aplicațiilor		4	
Bibliografie 1. Documentation for app developers, Google LLC, https://developer.android.com/docs 2. Angular Docs, Google LLC, https://angular.io/docs 3. .NET documentation, Microsoft Corporation, https://docs.microsoft.com/en-us/dotnet/			
8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore	Observații
Introducere și organizare Prezentarea de ansamblu a desfășurării laboratorului Teme abordate și cunoștințe dobândite la finalul semestrului Prezentarea baremului de notare	Învățare prin descoperire Dialog Explicație Dezbatere Proiect Implementare Lucru în echipa	2	
Setarea mediului de lucru Prezentarea tehnosistemului de lucru Instalarea și configurarea instrumentelor necesare		2	
Arhitectura aplicațiilor și componentele esențiale Arhitectura unei aplicații web/mobile Lucrul cu componentele cheie client/server		2	
Programare asincronă și multitasking Lucrul cu mai multe fire de execuție Async/await, thread-uri sau workers		2	
Laborator 6: Comunicare client-server Stabilirea canalului de comunicare între client și server API REST, WebSocket (opțional), teste simple de conectare		2	
Sistem de autentificare Crearea unui sistem de login/autentificare Gestionarea sesiunilor, tokeni (JWT), criptare parole		2	
Lucrul cu baze de date Modelarea datelor, conectarea la BD Mapare obiect-relațională, operații CRUD		2	
Interfață grafică și UI Libraries Aprofundarea elementelor de interfață grafică Lucru cu formulare, validări, folosirea unei biblioteci UI		2	
Integrare și servire aplicație completă Servirea aplicației client din server Integrarea componentelor: frontend, backend, BD Deploy local sau pe platforme gratuite		4	
Bibliografie			

1. Documentation for app developers, Google LLC, https://developer.android.com/docs
2. Angular Docs, Google LLC, https://angular.io/docs
3. .NET documentation, Microsoft Corporation, https://docs.microsoft.com/en-us/dotnet/

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, ale asociaților profesionale şi ale angajatorilor reprezentativi din domeniul aferent programului.

Fișa disciplinei a fost elaborată după consultări cu partenerii din mediul academic și economic ai Facultății de Matematică și Informatică.

10. Evaluare

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Formarea competențelor vizate prin conținutul disciplinei și dovedite prin realizarea documentației proiectului (obligatorie)	1. proba practica 2. proba orala 3. proiect - susținerea prezentării însoțite de un demo pentru soluția implementată, cu implicarea tuturor membrilor echipei de proiect	100%
10.5 Seminar/ laborator/ proiect	• Respectarea specificațiilor stabilite pentru dezvoltarea proiectului • Corectitudinea răspunsurilor la întrebările privind implementarea proiectului		
10.6 Standard minim de performanță			
Pentru promovarea examenului este necesara: • întocmirea documentației proiectului la nivelul descris în prezentarea activității pe disciplina (pagina e-learning a disciplinei) • implementarea și prezentarea proiectului, cu respectarea cerințelor și termenelor stabilite în prezentarea activității pe disciplina (pagina e-learning a disciplinei).			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. dr. Ion Gabriel Stan	Director de departament Conf. dr. Nicușor Minculete
Titular de curs Conf. dr. Alexandra Băicoianu Gabriel Alexandru	Titular de laborator Gabriel Alexandru

Notă:

1) Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

2) Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

3) Regimul disciplinei (conținut) - se alege una din variantele**: DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

4) Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

5) Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea *Transilvania* din Braşov
1.2 Facultatea	Matematică şi Informatică
1.3 Departamentul	Matematică şi Informatică
1.4 Domeniul de studii	Informatică
1.5 Ciclul de studii[1)]	Licență
1.6 Programul de studii/ Calificarea	Informatică Aplicată / Licențiat în Informatică

2. Date despre disciplină

2.1 Denumirea disciplinei	**MODELARE ȘI ORCHESTRARE CU LLM-URI ȘI AGENȚI AI**							
2.2 Titularul activităților de curs				Conf. Dr. Alexandra Băicoianu				
2.3 Titularul activităților de seminar/ laborator/ proiect				Drd. Bogdan Petre				
2.4 Anul de studiu	III	2.5 Semestrul	II	2.6 Tipul de evaluare	V	2.7 Regimul disciplinei	Conținut[2)]	DS
							Obligativitate[3)]	DO

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

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs		2	3.3 seminar/ laborator/ proiect	0/2/0
3.4 Total ore din planul de învățământ	40	din care: 3.5 curs		20	3.6 seminar/ laborator/ proiect	0/20/0

	Distribuția fondului de timp	ore
	Studiul după manual, suport de curs, bibliografie şi notițe	38
	Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	15
	Pregătire seminarii/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri	45
	Tutoriat	6
	Examinări	6
	Alte activități:.....	
	110	

3.7 Total ore studiu individual	*110*	
3.8 Total ore pe semestru	*150*	
3.9 Numărul de credite	*5*	

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Noțiuni de bază în programare (Python). • Familiaritate cu rețele neuronale și învățare automată (nivel introductiv). • Cunoașterea structurilor de date și algoritmilor fundamentali.
4.2 de competențe	• Abilități de lucru cu medii de dezvoltare (VS Code, Jupyter Notebook). • Competențe de bază în prelucrarea datelor.

5. Condi ii (acolo unde este cazul)

5.1 de desfăşurare a cursului	Sală de minim 150 de locuri, videoproiector
5.2 de desfăşurare a seminarului/ laboratorului/ proiectului	Acces la calculatoare cu Python 3.x, biblioteci TensorFlow, PyTorch, LangChain). Conexiune la internet pentru utilizarea modelelor LLM.

6. Competen e specifice acumulate

Competențe profesionale	CP1. Programarea în limbaje de nivel înalt **Cunoștințe** R.Î.1.1. Absolventul poate să scrie cod clar și bine structurat în limbaje de programare specifice. R.Î.1.2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale. **Aptitudini** R.Î.1.3. Absolventul poate să utilizeze biblioteci și framework-uri pentru îmbunătățirea performanțelor și funcționalitatea aplicațiilor software. R.Î.1.4. Absolventul poate să dezvolte aplicații complexe cu cerințele propuse de către utilizatori. **Responsabilitate și autonomie** R.Î.1.5. Absolventul este capabil să organizeze, planifice și supravegheze dezvoltarea aplicațiilor și a cadrelor necesare pentru a crea un produs software complet, de la etapele inițiale de planificare până la testarea finală. CP2. Gestionarea ciclului de viață al aplicațiilor informatice **Cunoștințe** R.Î.2.1. Absolventul poate să parcurgă ciclul de viață al unui sistem informatic, începând cu ancheta preliminară și terminând cu exploatarea și menținerea sistemului. **Aptitudini** R.Î.2.3. Absolventul poate să proiecteze soluții software eficiente și scalabile pentru fiecare fază a ciclului de viață. R.Î.2.4. Absolventul poate să implementeze și să testeze componente software conform specificațiilor funcționale și non-funcționale. R.Î.2.5. Absolventul poate să monitorizeze și să optimizeze performanța sistemului în faza de exploatare. **Responsabilitate și autonomie** R.Î.2.8. Absolventul poate să lucreze în echipă pentru a parcurge ciclul de viață al unui sistem informatic. CP3. Utilizarea instrumentelor informatice în context interdisciplinar **Cunoștințe** R.Î.3.1. Absolventul poate utiliza instrumente informatice (folosind tehnologii specifice) pentru a colecta, analiza și interpreta date din diferite domenii de activitate. R.Î.3.2. Absolventul poate evalua și selecta instrumente informatice adecvate pentru rezolvarea problemelor interdisciplinare, având abilitatea de a analiza nevoile specifice ale unui proiect și de a alege soluții tehnologice eficiente. **Aptitudini** R.Î.3.4. Studentul/absolventul dezvoltă interesul pentru literatura de specialitate națională și internațională, participarea la cursuri specializate, conferințe, simpozioane, etc. pentru dezvoltarea profesională şi personală continuă și adaptarea eficientă la noile descoperiri ştiin ifice. **Responsabilitate și autonomie** R.Î.3.5. Absolventul poate să comunice și să colaboreze eficient cu specialiști din alte domenii, utilizând instrumente informatice specifice, dezvoltând competențe de lucru în echipă și abilități de gestionare a proiectelor interdisciplinare.

Competențe transversale	CT1 Aplicarea regulilor de muncă organizată şi eficientă, a unor atitudini responsabile față de domeniul didactic-ştiințific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor şi a normelor de etică profesională R.Î.1.1. Absolventul este capabil să-și organizeze și gestioneze timpul și resursele pentru a atinge obiectivele de învățare. R.Î.1.2. Absolventul este capabil să desfășoare activitate creatoare, să se dezvolte profesional și să abordeze noi domenii, adaptându-se cerințelor nou apărute. CT2 Desfăşurarea eficientă a activităților organizate într-un grup inter-disciplinar şi dezvoltarea capacităților empatice de comunicare inter-personală, de relaționare şi colaborare cu grupuri diverse R.Î.2.1. Absolventul va putea utiliza instrumente și tehnici de comunicare pentru a rezolva probleme și a dezvolta soluții inovatoare. R.Î.2.2. Absolventul va relaționa și colabora cu grupuri diverse, inclusiv cu persoane din alte culturi sau medii socio-economice. CT3 Utilizarea unor metode şi tehnici eficiente de învățare, informare, cercetare şi dezvoltare a capacităților de valorificare a cunoştințelor, de adaptare la cerințele unei societăți dinamice şi de comunicare în limba română şi într-o limbă de circulație internațională R.Î.3.1. Absolventul va utiliza metode și tehnici eficiente de învățare pentru a-și îmbunătăți performanța academică. R.Î.3.2. Absolventul va utiliza instrumente și tehnici eficiente pentru a informa și comunica informații complexe într-un mod clar și concis.

7. Obiectivele disciplinei (reieşind din grila competen elor specifice acumulate)

7.1 Obiectivul general al disciplinei	Obiectivul principal al disciplinei este dobândirea competențelor teoretice și practice necesare pentru proiectarea, dezvoltarea și evaluarea sistemelor inteligente bazate pe modele de limbaj de mari dimensiuni (LLM) și agenți AI autonomi. Studenții vor învăța să înțeleagă principiile de funcționare ale acestor tehnologii, să identifice aplicații relevante în diverse domenii și să dezvolte soluții concrete folosind instrumente moderne și bune practici din industrie. Accentul va fi pus atât pe partea tehnică (arhitectură, integrare, testare), cât și pe aspectele critice legate de etică, responsabilitate și impactul socio-tehnologic al acestor sisteme.
7.2 Obiectivele specifice	• Însușirea conceptelor de bază ale LLM-urilor și transformatoarelor. • Implementarea practică a modelelor open-source (DeepSeek, Mistral). • Dezvoltarea de agenți AI cu funcționalități complexe (RAG, execuție de acțiuni). • Evaluarea critică a performanțelor și limitelor modelelor.

8. Conținuturi

8.1 Curs	Metode de predare	Observații
Introducere în Generative AI și LLM-uri: Definiții, aplicații, comparație cu NLP tradițional		2 ore

Arhitectura transformatoarelor: Mecanisme de autoatentie, componente, normalizare	Prelegeri	2 ore
Modele populare: GPT, BERT, arhitecturi encoder-decoder		2 ore
LLM-uri open-source: DeepSeek, LLaMA, Mistral	Expuneri	2 ore
Prompt engineering: Tehnici pentru interacțiune eficientă	Dialog	2 ore
Construirea agenților AI: LangChain, RAG, sisteme cu memorie		4 ore
Evaluare: Benchmark-uri (GLUE, SQUAD), metrici	Problematizare	2 ore
Studii de caz: Aplicații în analiza de sentiment, chatbots	Studiu de caz	2 ore
Dezvoltare etică și responsabilă: Limitări, riscuri, bune practici	Exemple	2 ore
		Total 20 de ore

Bibliografie:

- "Natural Language Processing with Transformers" – Tunstall, von Werra, Wolf, O'Reilly, 2022
- "Deep Learning" – Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press, 2016
- "Prompt Engineering for Generative AI" – James Phoenix, O'Reilly, 2023
- "Mastering LLMs with Hugging Face" – Lewis Tunstall, Packt, 2023
- "AI Agents with LangChain" – Harrison Chase, O'Reilly, 2023
- "Building LLM-Powered Applications" – Valentina Alto, Manning, 2024

8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Observații
Configurarea mediului. Testare modele preantrenate pe sarcini simple.		2 ore
Prelucrarea datelor. Curățare și tokenizare text. Încărcare și adaptare a seturilor de date		3 ore
Fine-tuning pe sarcini simple. Antrenare pe clasificare binară. Evaluare cu metrici	Exerciții	3 ore

LLM-uri open-source. Comparare între LLaMA, Mistral, DeepSeek. Generare de text și completare de propoziții	Dialog Lucru în echipă Problematizare Studiu individual	2 ore
Prompt engineering. Optimizarea prompturilor pentru sarcini specifice		2 ore
RAG (Retrieval-Augmented Generation). Integrare cu vector databases (FAISS). Construire sistem de Q&A pe documente proprii		4 ore
Agenți AI cu LangChain. Creare de agenți simpli cu memorie. Legături la API-uri externe		4 ore
		Total 20 de ore

Bibliografie:

- "Natural Language Processing with Transformers" – Tunstall, von Werra, Wolf, O'Reilly, 2022
- "Deep Learning" – Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press, 2016
- "Prompt Engineering for Generative AI" – James Phoenix, O'Reilly, 2023
- "Mastering LLMs with Hugging Face" – Lewis Tunstall, Packt, 2023
- "AI Agents with LangChain" – Harrison Chase, O'Reilly, 2023
- "Building LLM-Powered Applications" – Valentina Alto, Manning, 2024

9. Coroborarea conținuturilor disciplinei cu aşteptările reprezentanților comunităților epistemice, asociațiilor profesionale şi angajatori reprezentativi din domeniul aferent programului

Coroborarea se aplică în *Acordurile de parteneriat* şi în *Conven iile de Practică* încheiate cu partenerii din mediul socio-economic ai Facultății/Universității.

10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
10.4 Proiect	Formarea competențelor vizate prin conținutul disciplinei. Atingerea obiectivelor educaționale din fișa disciplinei.	Evaluarea finală constă în realizarea unui proiect aplicat, care poate fi dezvoltat individual sau în echipe de cel mult 3 studenți. Proiectul presupune implementarea unui sistem funcțional bazat pe modele de limbaj de mari dimensiuni (LLM) și/sau agenți AI, cu aplicabilitate practică într-un domeniu ales (ex: educație, sănătate, business, asistenți inteligenți etc.). Livrabilele proiectului vor include: implementarea efectivă a sistemului (cu cod sursă); documentarea completă a arhitecturii și a fluxului de lucru; analiza și interpretarea rezultatelor obținute în urma testării	100%

		sistemului; discuții critice privind limitările, avantajele și posibile direcții de îmbunătățire; o prezentare finală susținută în fața colegilor și a cadrului didactic, însoțită de o demonstrație practică a funcționării proiectului. Proiectul final are caracter eliminatoriu - promovarea cursului este condiționată de realizarea și prezentarea acestuia în termenii și la standardul cerut.	
10.5 Standard minim de performan ă			
Înțelegerea și aplicarea conceptelor de bază ale LLM-urilor, arhitectura transformatoarelor, diferențierea între modele Utilizarea unui LLM open-source pentru o sarcină simplă Aplicarea unor tehnici de preprocesare pe un set de date pentru NLP. Calculul metricilor de bază (ex: perplexity, BLEU) pe un set de date. Compararea performanței a două modele (ex: LLaMA vs. GPT-2) pe o sarcină comună.			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 17/07/2025 și aprobată în ședința de Consiliu al facultății din data de 17/07/2025.

Decan Conf. Dr. Ion Gabriel Stan	Director de departament Conf. Dr. **Nicușor Minculete**
Titular de curs Conf. Dr. Alexandra Băicoianu	Titular de laborator Drd. Bogdan Petre

Notă:

[1)] Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor şi al specializărilor/ programelor de studii universitare în vigoare);

[2)] Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

[3)] Regimul disciplinei (conținut) - se alege una din variantele: **DF** (disciplină fundamentală)/ **DD** (disciplină din domeniu)/ **DS** (disciplină de specialitate)/ **DC** (disciplină complementară) - pentru nivelul de licență; **DAP** (disciplină de aprofundare)/ **DSI** (disciplină de sinteză)/ **DCA** (disciplină de cunoaştere avansată) - pentru nivelul de masterat;

[4)] Regimul disciplinei (obligativitate) - se alege una din variantele: **DI** (disciplină obligatorie)/ **DO** (disciplină opțională)/ **DFac** (disciplină facultativă);

[5)] Un credit este echivalent cu 30 de ore de studiu (activități didactice şi studiu individual).