FIŞA DISCIPLINEI

Aplicaţii ale inteligenţei computaţionale în ingineria software

Anul universitar 2025-2026

1. Date despre program

1.1. Instituţia de învăţământ superior	Universitatea Babeş-Bolyai Cluj-Napoca
1.2. Facultatea	Facultatea de Matematică şi Informatică
1.3. Departamentul	Departamentul de informatică
1.4. Domeniul de studii	Informatică
1.5. Ciclul de studii	Master
1.6. Programul de studii / Calificarea	Inginerie Software
1.7. Forma de învăţământ	Cu frecvenţă

2. Date despre disciplină

2.1. Denumirea disciplinei	Aplicaţii ale inteligenţei computaţionale în ingineria soft					Codul disciplinei	MME8063
2.2. Titularul activităţilor de curs	Prof. dr. Czibula Istvan						
2.3. Titularul activităţilor de seminar	Prof. dr. Czibula Istvan						
2.4. Anul de studiu	2	2.5. Semestrul	3	2.6. Tipul de evaluare	E	2.7. Regimul disciplinei	Obligatoriu

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

3.1. Număr de ore pe săptămână	4	din care: 3.2. curs	2	3.3. seminar/ laborator/proiect	1 sem + 1 pr
3.4. Total ore din planul de învăţământ	56	din care: 3.5. curs	28	3.6 seminar/laborator/proiect	28
Distribuţia fondului de timp pentru studiul individual (SI) şi activităţi de autoinstruire (AI)					**ore**
Studiul după manual, suport de curs, bibliografie şi notiţe (AI)					26
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren					36
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii şi eseuri					35
Tutoriat (consiliere profesională)					12
Examinări					10
Alte activităţi					
3.7. Total ore studiu individual (SI) şi activităţi de autoinstruire (AI)				119	
3.8. Total ore pe semestru				175	
3.9. Numărul de credite				7	

4. Precondiţii (acolo unde este cazul)

4.1. de curriculum	Inteligenţă Artificială
4.2. de competenţe	Abilităţi de programare

5. Condiţii (acolo unde este cazul)

5.1. de desfăşurare a cursului	Sală de curs cu videproiector
5.2. de desfăşurare a seminarului/ laboratorului	Laborator cu calculatoare; medii de dezvoltare software (.NET, Java, etc)

6.1. Competenţele specifice acumulate[1]

Competenţe profesionale/esenţiale	• înţelegerea şi operarea cu conceptele de bază din domeniul ingineriei software; • capacitatea de analiză şi sinteză a informaţiei; • modelare şi rezolvarea de probleme din lumea reală;
Competenţe transversale	• comportarea onorabilă, etică, respectarea deontologiei profesionale; • abilităţi de muncă în echipă, cu preluarea diferitelor roluri de execuţie şi conducere pentru realizarea unor proiecte.

6.2. Rezultatele învăţării

Cunoştinţe	Studentul cunoaşte: • norme şi reguli etice şi deontologice în cercetarea ştiinţifică; • cerinţele specifice demersului de cercetare în domeniul informaticii în general şi al ingineriei software în special şi înţelege rolul cercetării în promovarea progresului; • concepte fundamentale de modelare prin care analizează probleme din viaţa reală, le transpune în cerinţe concrete şi elaborează un model software corespunzător
Aptitudini	Studentul este capabil să: • aibă o viziune interdisciplinară între diferite subdomenii ale informaticii şi de a le combina într-un sistem software; • conceapă, modeleze şi proiecteze aplicaţii software complexe.
Responsabilităţi şi autonomie	Studentul are capacitatea de a lucra independent pentru: • obţinerea informaţiilor necesare proiectării, organizării, realizării şi evaluării demersurilor de cercetare în inginerie software; • a concepe, modela şi proiecta aplicaţii software complexe.

7. Obiectivele disciplinei (reieşind din grila competenţelor acumulate)

7.1 Obiectivul general al disciplinei	Prezentarea domeniului Ingineriei software bazat pe căutare (SBSE) ca o nouă direcţie de cercetare şi aplicare în domeniul ingineriei software.
7.2 Obiectivele specifice	• Introducerea domeniului SBSE. • Imtroducerea necesităţii şi a importaţei utilizării tehnicilor inteligenţei computaţionale în rezolvarea problemelor din domeniul ingineriei software. • Prezentarea unor activotăţi importante din dezvoltarea software şi cum pot fi acestea rezolvate folosind tehnici ale inteligenţei computaţionale.

[1] Se poate opta pentru competenţe sau pentru rezultatele învăţării, respectiv pentru ambele. În cazul în care se alege o singură variantă, se va şterge tabelul aferent celeilalte opţiuni, iar opţiunea păstrată va fi numerotată cu 6.

8. Conţinuturi

8.1 Curs	Metode de predare	Observaţii
1. Introducere • Ingineria software bazată pe căutare (Search Based Software Engineering – SBSE) • Concepte de bază şi probleme abordate	• Expunerea interactivă • Explicaţia • Conversaţia • Demonstraţia didactică	
2. Instruire automată în ingineria software ▪ Tehnici de instruire automată • Aplicaţii	• Expunerea interactivă • Explicaţia • Conversaţia • Demonstraţia didactică	
3. Tehnici ale inteligenţei computaţionale (CI) pentru înţelegerea programelor	• Expunerea interactivă • Explicaţia • Conversaţia • Demonstraţia didactică	
4. Tehnici CI pentru refactorizare	• Expunerea interactivă • Explicaţia • Conversaţia • Demonstraţia didactică	
5. Tehnici CI pentru detectarea şi predicţia defectelor software	• Expunerea interactivă • Explicaţia • Conversaţia • Demonstraţia didactică	
6. Tehnici CI pentru testare software	• Expunerea interactivă • Explicaţia • Conversaţia • Demonstraţia didactică	
7. Tehnici CI pentru pentru vizualizare software	• Expunerea interactivă • Explicaţia • Conversaţia • Demonstraţia didactică	
8. Tehnici CI pentru estimarea costului si predicţia efortului de dezvoltare software	• Expunerea interactivă • Explicaţia • Conversaţia • Demonstraţia didactică	
9. Tehnici CI pentru reutilizare software	• Expunerea interactivă • Explicaţia • Conversaţia • Demonstraţia didactică	
10. Tehnici CI pentru identificarea şabloanelor de proiectare	• Expunerea interactivă • Explicaţia • Conversaţia • Demonstraţia didactică	
Prezentare rapoarte de cercetare CISE	• Expunerea interactivă • Conversaţia • Evaluarea orală	

Bibliografie

1. Czibula, I., G., Use of search techniques to software development, Editura Risoprint, ISBN 978-973-53- 0119-4, 2009 (248 pagini)
2. Mark Harman and Bryan F. Jones. Search-based software engineering. Information & Software Technology, 43(14):833-839, 2001.
3. Olaf Seng, Johannes Stammel, and David Burkhart. Search-based determination of refactorings for improving the class structure of object-oriented systems. In GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation, pages 1909{1916, New York, NY, USA, 2006. ACM Press.
4. Frank Simon, Frank Steinbruckner, and Claus Lewerentz. Metrics based refactoring. In CSMR '01: Proceedings of the Fifth European Conference on Software Maintenance and Reengineering, pages 30-38, Washington, DC, USA, 2001. IEEE Computer Society

8.2 Seminar/laborator	Metode de predare	Observații
1. Aspecte administrative.	• Expunerea interactivă • Explicația • Conversația	Activitatea este structurată sub forma a 2 ore din 2 în 2 săptămâni.
2. Documentare pe Internet și Intranet în vederea alegerii temei pentru raportul de cercetare și alegerea săptămânii pentru prezentare.	• Documentare • Explicația • Conversația	
Un proiect software pe o team din SBSE (Proiect 1) va fi dezvoltat foosind un mediu de dezvoltare ML existent. Al doilea proiect software (Proiect 2) va fi dezvoltat și implementat. Aplicația va demonstra utilizarea tehnicilor CI pentru o sarcină specifică din ingineria software.		
3. Definirea și spcificarea problemei pentru Proiect 2	• Temă de laborator • Explicația • Conversația	
4. Comentarii despre soluție (analiza problemei) și modelarea din persectiva SBSE a problemei (Proiect 2). Demonstrarea Proiectului 1	• Temă de laborator • Explicația • Conversația	
5. Documentatția de proiectare a Proiectului 2	• Temă de laborator • Explicația • Conversația	
6. Codul sursă pentru Proiectul 2. Demonstrarea proiectului 2	• Temă de laborator • Explicația • Conversația	

Bibliografie

1. Czibula, I., G., Use of search techniques to software development, Editura Risoprint, ISBN 978-973-53- 0119-4, 2009 (248 pagini)
2. Mark Harman and Bryan F. Jones. Search-based software engineering. Information & Software Technology, 43(14):833-839, 2001.
3. Olaf Seng, Johannes Stammel, and David Burkhart. Search-based determination of refactorings for improving the class structure of object-oriented systems. In GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation, pages 1909{1916, New York, NY, USA, 2006. ACM Press.
4. Frank Simon, Frank Steinbruckner, and Claus Lewerentz. Metrics based refactoring. In CSMR '01: Proceedings of the Fifth European Conference on Software Maintenance and Reengineering, pages 30- 38, Washington, DC, USA, 2001. IEEE Computer Society.

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

Conținutul disciplinei este consistent cu cel al disciplinelor similar de la alte universități din țară și din străinătate, precum și cu cerințele pe care le-ar avea potențiali angajatori în domeniul Software Engineering.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	• Va fi redactat și prezentat un raport de cercetare pe o tema din domeniul SBSE, pe baza unor rezultate recente obținute în cercetare	Evaluarea raportului de cercetare (articolul scris și prezentarea orală)	20%
	• Corectitudinea și completitudinea cunoștințelor acumulate	Examen scris (în sesiune)	40%
	• Activitate	4 absențe nemotivate sunt premise, restul se vor depuncta	10%
10.5 Seminar/Laborator	• Realizarea unui proiect software SBSE folosind un software ML open • source	Evaluare proiectului (documentare și demonstrare)	15%
	• Un proiect software din domeniul SBSE va fi implementat în întregime, fără a folosi biblioteci ML.	Evaluarea proiectului (implementare, documentare și demonstrare)	15%
10.6 Standard minim de performanță			
• Cunoașterea elementelor fundamentale de teorie. Fiecare student trebuie să demonstreze că a atins un nivel acceptabil de cunoaștere și înțelegere a domeniului SBSE, că este capabil să exprime cunoștințele într-o formă coerentă, că are capacitatea de a stabili anumite conexiuni și de a utiliza cunoștințele în rezolvarea unor probleme. • Pentru promovare e necesar ca nota finală să fie minimum 5.			

11. Etichete ODD (Obiective de Dezvoltare Durabilă / Sustainable Development Goals)[2]

Nu se aplică.

Data completării:	Semnătura titularului de curs	Semnătura titularului de seminar
.		
10.04.2025	Prof. dr. Istvan Gergely Czibula	Prof. dr. Istvan Gergely Czibula

Data avizării în departament: Semnătura directorului de departament

Conf. dr. Adrian STERCA

[2] Păstrați doar etichetele care, în conformitate cu *Procedura de aplicare a etichetelor ODD în procesul academic*, se potrivesc disciplinei și ștergeți-le pe celelalte, inclusiv eticheta generală pentru *Dezvoltare durabilă* - dacă nu se aplică. Dacă nicio etichetă nu descrie disciplina, ștergeți-le pe toate și scrieți "*Nu se aplică.*".