

FIȘA DISCIPLINEI

Proiect: Proiectarea algoritmilor si sistemelor software

Anul universitar 2025

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Babeș-Bolyai Cluj-Napoca
1.2. Facultatea	Facultatea de Matematică și Informatică
1.3. Departamentul	Departamentul de Informatică
1.4. Domeniul de studii	Calculatoare și Tehnologia Informației
1.5. Ciclu de studii	Licență
1.6. Programul de studii / Calificarea	Ingineria Informației (în limba engleză)
1.7. Forma de învățământ	Cu frecvență

2. Date despre disciplină

2.1. Denumirea disciplinei		Project: Proiectarea algoritmilor si sistemelor software				Codul disciplinei	MLE5192
2.2. Titularul activităților de curs		Asist. Dr. Horea-Bogdan Mureșan					
2.3. Titularul activităților de seminar		Asist. Dr. Horea-Bogdan Mureșan					
2.4. Anul de studiu	2	2.5. Semestrul	4	2.6. Tipul de evaluare	C	2.7. Regimul disciplinei	Obligativu

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

3.1. Număr de ore pe săptămână	2	din care: 2. curs	-	3.3. seminar/ laborator/proiect	0/0/2
3.4. Total ore din planul de învățământ	28	din care: 35. curs	-	3.6 seminar/laborator/proiect	28
Distribuția fondului de timp pentru studiul individual (SI) și activități de autoinstruire (AI)					ore
Studiul după manual, suport de curs, bibliografie și notițe (AI)					6
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					6
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					6
Tutoriat (consiliere profesională)					-
Examinări					4
Alte activități					-
3.7. Total ore studiu individual (SI) și activități de autoinstruire (AI)				22	
3.8. Total ore pe semestru				50	
3.9. Numărul de credite				2	

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	• Metode avansate de programare • Baze de date • Sisteme de operare
4.2. de competențe	• Abilitatea de a programa într-un limbaj de nivel înalt • Concepte de baza despre baze de date • Concepte de baza despre rețele de calculatoare

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	• Proiector
5.2. de desfășurare a seminarului/ laboratorului	• Laboratoare cu calculatoare

	• Medii de programare pentru Java si .NET • Sisteme de gestiune a bazelor de date
--	--

6.1. Competențele specifice acumulate¹

Competențe profesional e/esențiale	C2.1 Descrierea structurii și funcționării componentelor hardware, software și de comunicații C2.2 Explicarea rolului, interacțiunii și funcționării componentelor sistemelor hardware, software și de comunicații C2.3 Construirea unor componente hardware, software și de comunicații folosind metode de proiectare, limbaje, algoritmi, structuri de date, protocoale și tehnologii C2.4 Evaluarea caracteristicilor funcționale și nefuncționale ale componentelor hardware, software și de comunicații, pe baza unor metrici C2.5 Implementarea componentelor sistemelor hardware, software și de comunicație C4.1 Identificarea și descrierea tehnologiilor și mediilor de programare și ale conceptelor specifice ingineriei programării C4.2 Explicarea rolului, interacțiunii și funcționării componentelor sistemelor software C4.3 Elaborarea specificațiilor și proiectarea unor sisteme informatice folosind metode și instrumente specifice C4.4 Gestionarea ciclului de viață a sistemelor hardware, software și de comunicații pe baza evaluării performanțelor C4.5 Dezvoltarea și implementarea și integrarea soluțiilor software
Competențe transversale	CT1 Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesiei. CT2 Identificarea, descrierea și derularea proceselor din managementul proiectelor, cu preluarea diferitelor roluri în echipă și descrierea clară și concisă, verbal și în scris, a rezultatelor din domeniul de activitate. CT3 Demonstrarea spiritului de inițiativă și acțiune pentru actualizarea cunoștințelor profesionale, economice și de cultură organizațională.

6.2. Rezultatele învățării

Cunoștințe	• Absolventul are cunoștințe adecvate legate de folosirea mediilor de dezvoltare integrate în scopul creării de aplicații complexe de dimensiuni mari. • Absolventul are cunoștințe necesare pentru utilizarea calculatoarelor, dezvoltarea programelor și aplicațiilor software, procesarea informațiilor.
Aptitudini	• Absolventul este capabil să aplice șabloane arhitecturale, șabloane de proiectare și bunele practici în domeniu pentru a proiecta aplicații software de complexitate mare. • Absolventul este capabil să combine informații diverse pentru a formula soluții și dezvolta idei de dezvoltare pentru noi produse și aplicații.
Responsabilități și autonomie	• Absolventul are capacitatea de a alege și folosi paradigme de programare (procedural, orientat obiect, funcțional) pentru realizarea de aplicații software adecvate specificului domeniului aplicației dezvoltate. • Absolventul are cunoștințele necesare legate de etapele ciclului de viață al softului și a modelelor de procese software.

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	• Înțelegerea conceptelor și problemelor sistemelor distribuite • Competențe îmbunătățite de proiectare și programare
--	--

¹ Se poate opta pentru competențe sau pentru rezultatele învățării, respectiv pentru ambele. În cazul în care se alege o singură variantă, se va șterge tabelul aferent celeilalte opțiuni, iar opțiunea păstrată va fi numerotată cu 6.

7.2 Obiectivele specifice	• Insusirea conceptelor si tehnicilor existente pentru dezvoltarea aplicațiilor distribuite. • Familiarizarea cu concepte moderne din dezvoltarea sistemelor soft.
----------------------------------	---

8. Conținuturi

8.1 Curs	Metode de predare	Observații
Bibliografie 1. Joseph Albahari and Ben Albahari, C# 6.0 in a Nutshell, Sixth Edition, O'Reilly, 2015. 2. Larman, C.: Applying UML and Design Patterns: An Introduction to OO Analysis and Design and Unified Process, Berlin, Prentice Hall, 2002. 3. Fowler, M., Patterns of Enterprise Application Architecture, Addison-Wesley, 2002. 4. Hohpe, G., Woolf, B., Enterprise integration patterns, Addison-Wesley, 2003. 5. ***, Microsoft Developer Network, Microsoft Inc., http://msdn.microsoft.com/ 6. ***, The Java Tutorial, SUN Microsystems, Inc. http://download.oracle.com/javase/tutorial/ 7. Eckel, B., Thinking in Java, 4th edition, Prentice Hall, 2006 9. Walls, Craig, Spring in Action, Fourth Edition, Ed. O'Reilly, 2015. 10. Documentație Spring http://projects.spring.io/spring-framework/		
8.2 Seminar / laborator	Metode de predare	Observații
S1. Folosirea unui instrument de construire automata. Alegerea aplicatiei pentru proiect.	Conversatia, studii de caz, evaluarea	
S2. Accesarea unei baze de date relationale.	Conversatia, studii de caz, evaluarea	
S3. Configurarea unei aplicații folosind IoC.	Conversatia, studii de caz, evaluarea	
S4-S5. Proiectarea și implementarea serviciilor (Șablonul Proxy).	Conversatia, studii de caz, evaluarea	
S7. Enterprise Application Integration (Protobuf, gRPC, Thrift).	Conversatia, studii de caz, evaluarea	
S8. Instrumente ORM.	Conversatia, studii de caz, evaluarea	
S9. Servicii REST.	Conversatia, studii de caz, evaluarea	
S10. Clienti Web.	Conversatia, studii de caz, evaluarea	
S11. Asynchronous messaging systems.	Conversatia, studii de caz, evaluarea	
S12. Websockets.	Conversatia, studii de caz, evaluarea	

S13. Securitate servicii REST	Conversatia, studii de caz, evaluarea	
Bibliografie 1. Joseph Albahari and Ben Albahari, C# 6.0 in a Nutshell, Sixth Edition, O'Reilley, 2015. 2.***, Microsoft Developer Network, Microsoft Inc., http://msdn.microsoft.com/ 3.***, The Java Tutorial, SUN Microsystems, Inc. http://download.oracle.com/javase/tutorial/ 4. Walls, Craig, Spring in Action, Fourth Edition, Ed. O'Reilley, 2015. 5. Documentatie Spring http://projects.spring.io/spring-framework/		

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

- Cursul respectă recomandările curriculare IEEE și ACM pentru studiile în informatică
- Companiile de software consideră conținutul cursului ca fiind util în dezvoltarea abilităților de modelare și programare ale studenților

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs			
10.5 Seminar/laborator	Capacitatea de a proiecta și dezvolta diferite tipuri de aplicații distribuite.	Teme, sisteme dezvoltate, documentații	100%
10.6 Standard minim de performanță			
• Pentru a promova disciplina studentul trebuie să obțină cel puțin nota 5 la temele de laborator.			

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

Nu se aplică.

Data completării:
14.04.2025

Semnătura titularului de curs
Asist. Dr. Horea-Bogdan Mureșan

Semnătura titularului de seminar
Asist. Dr. Horea-Bogdan Mureșan

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".