

FIȘA DISCIPLINEI

Design Patterns

Anul universitar 2025 - 2026

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Babeș-Bolyai din Cluj-Napoca
1.2. Facultatea	Facultatea de Matematică și Informatică
1.3. Departamentul	Departamentul de Informatică
1.4. Domeniul de studii	Informatică
1.5. Ciclul de studii	Licență
1.6. Programul de studii / Calificarea	Informatică
1.7. Forma de învățământ	Cu frecvență

2. Date despre disciplină

2.1. Denumirea disciplinei	Design Patterns	Codul disciplinei	MLE8115				
2.2. Titularul activităților de curs	Conf. Univ. Dr. Molnar Arthur-Jozsef						
2.3. Titularul activităților de seminar	Conf. Univ. Dr. Molnar Arthur-Jozsef						
2.4. Anul de studiu	3	2.5. Semestrul	6	2.6. Tipul de evaluare	C	2.7. Regimul disciplinei	Opțională

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

3.1. Număr de ore pe săptămână	3	din care: 3.2. curs	2	3.3. seminar/ laborator/proiect	1
3.4. Total ore din planul de învățământ	36	din care: 3.5. curs	24	3.6 seminar/laborator/proiect	12
Distribuția fondului de timp pentru studiul individual (SI) și activități de autoinstruire (AI)					ore
Studiul după manual, suport de curs, bibliografie și notițe (AI)					20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					20
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					30
Tutoriat (consiliere profesională)					14
Examinări					2
Alte activități					3
3.7. Total ore studiu individual (SI) și activități de autoinstruire (AI)				89	
3.8. Total ore pe semestru				125	
3.9. Numărul de credite				5	

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Fundamentele Programării, Programare Orientată Obiect
4.2. de competențe	Abilități de programare într-un limbaj care suportă paradigma programării orientate obiect (preferabil unul din limbajele Python, C++, Java sau C#)

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Sală de curs cu video-proiector și acces la Internet.
5.2. de desfășurare a seminarului/ laboratorului	Sală de curs cu video-proiector și acces la Internet.

6.1. Competențele specifice acumulate¹

Competențe profesionale/esențiale	<ul style="list-style-type: none">• Programarea în limbaje de nivel înalt• Dezvoltarea și întreținerea aplicațiilor informatice
Competențe transversale	<ul style="list-style-type: none">• Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională• 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

6.2. Rezultatele învățării

Cunoștințe	<ul style="list-style-type: none">• Absolventul are cunoștințe necesare pentru utilizarea calculatoarelor, dezvoltarea programelor și aplicațiilor software, procesarea informațiilor.• Absolventul este capabil să aplice șabloane arhitecturale, șabloane de proiectare și bunele practici în domeniu pentru a proiecta aplicații software de complexitate mare.• Absolventul are abilitatea de a înțelege și folosi șabloanele de proiectare pentru dezvoltarea aplicațiilor.
Aptitudini	<ul style="list-style-type: none">• Absolventul are aptitudinile necesare pentru conceperea programelor de calculator și analiza sistemelor software.• Absolventul are abilitatea de a aplica reguli generale unor probleme specifice și de a produce soluții relevante.
Responsabilități și autonomie	<ul style="list-style-type: none">• Absolventul are abilitatea de a înțelege și comunica eficient informațiile.• Absolventul este capabil să combine informații diverse pentru a formula soluții și genera idei de dezvoltare pentru noi produse și aplicații.• 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 este capabil de a prezenta și a explica metodele, algoritmi, paradigmele și tehnicile folosite în diferite ramuri ale informaticii.

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	Îmbunătățirea competențelor studenților legate de conceptele de proiectare software și șabloanele de proiectare prin utilizarea unei abordări pragmatice, empirice.
7.2 Obiectivele specifice	<ul style="list-style-type: none">• Facilitarea explorării diverselor limbaje care suportă paradigma dezvoltării orientate-obiect.• Crearea unui mediu în care cursanții pot explora utilizarea și utilitatea conceptelor de proiectare software în diverse scenarii de business.• Inducerea unei viziuni realiste și orientate către mediul comercial al conceptelor de proiectare și a aplicării șabloanelor de proiectare și a beneficiilor acestora.• Furnizarea unor modalități de lucru care au ca scop crearea de software de calitate.

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.

8. Conținuturi

8.1 Curs	Metode de predare
<p>Principiile programării orientate obiect (POO) Recapitulare ce acoperă principiile încapsulării, moștenirii, polimorfismului, coeziunii, cuplării, agregării și compoziției, specifice POO.</p>	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică
<p>Principiile SOLID Rolul lor de principii importante pentru software de calitate: principiul singurei responsabilități, principiul deschis-închis, substituția Liskov, segregarea interfețelor și inversarea dependențelor.</p>	
<p>Șabloane de proiectare creaționale (Abstract) Factory, Builder, Prototype, Singleton</p>	
<p>Șabloane de proiectare structurale Adapter, Bridge, Composite, Decorator, Façade, Flyweight, Proxy</p>	
<p>Șabloane de proiectare comportamentale Chain of responsibility, Command, Iterator, Mediator, Memento, Observer, State, Strategy, Template, Visitor.</p>	
<p>Anti-șabloane Abordări comune pentru probleme recurente care sunt de regulă ineficiente și riscă să fie contra-productive. O selecție de anti-șabloane cu exemple de cod care ilustrează pe unele din cele mai des întâlnite în codul sursă (ex., golden hammer, God class, spaghetti code)</p>	
<p>Șabloane "întunecate" în proiectarea experienței cu utilizatorul Șabloane înșelătoare al căror scop este de a păcăli utilizatorul pentru a întreprinde acțiuni pe care altfel nu le-ar face. O selecție de astfel de șabloane împreună cu exemple din aplicații existente (ex., drip pricing, roach motel, misdirection).</p>	
<p>Șabloane de proiectare de tipul Model-View-* O selecție de șabloane de proiectare de nivel mai înalt (Model View Controller, Model View ViewModel, Model View Presenter)</p>	
<p>Șabloane de tipul enterprise și arhitecturale O selecție de șabloane de nivel înalt întâlnite în aplicații enterprise ce necesită niveluri ridicate de disponibilitate și performanță (ex., message routing, pipes and filters, peer to peer, microservices).</p>	
<p>Bibliografie</p> <ol style="list-style-type: none"> 1. M. Fowler – <i>Patterns of Enterprise Application Architecture</i>, Addison Wesley, 2003 2. E. Freeman, E. Freeman, B. Bates – <i>Head-First Design Patterns</i>, Oreilly, 2004 3. E. Gamma, R. Helm, R. Johnson, J. Vlissides – <i>Design Patterns: Elements of Reusable Object-Oriented Software</i>, Addison Wesley, 1995 4. William J. Brown, Raphael C. Malveau, Hays W. "Skip" McCormick, Thomas J. Mowbray - <i>AntiPatterns: Refactoring Software, Architectures, and Projects in Crisis</i>, Wiley, 1998. 5. Harry Brignull - <i>Deceptive Patterns, Exposing the Tricks Tech Companies Use to Control You</i> (free edition on https://www.deceptive.design/book/contents/get-started), Testimonium Ltd., 2023. 	
8.2 Seminar / laborator	Metode de predare
Recapitularea conceptelor programării orientate obiect. Introducere în activitățile de laborator și modalitatea de notare.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică
Principiile SOLID. Șabloane de proiectare creaționale.	
Șabloane de proiectare structurale. Verificarea progresului proiectului de laborator.	
Șabloane de proiectare comportamentale. Verificarea progresului proiectului de laborator.	
Anti-șabloane. Șabloane întunecate de interacțiune cu utilizatorului. Șabloane arhitecturale și enterprise.	
Predarea proiectului de laborator.	
<p>Bibliografie</p> <ol style="list-style-type: none"> 1. M. Fowler – <i>Patterns of Enterprise Application Architecture</i>, Addison Wesley, 2003 2. E. Freeman, E. Freeman, B. Bates – <i>Head-First Design Patterns</i>, Oreilly, 2004 3. E. Gamma, R. Helm, R. Johnson, J. Vlissides – <i>Design Patterns: Elements of Reusable Object-Oriented Software</i>, Addison Wesley, 1995 4. William J. Brown, Raphael C. Malveau, Hays W. "Skip" McCormick, Thomas J. Mowbray - <i>AntiPatterns: Refactoring Software, Architectures, and Projects in Crisis</i>, Wiley, 1998. 5. Harry Brignull - <i>Deceptive Patterns, Exposing the Tricks Tech Companies Use to Control You</i> (free edition on https://www.deceptive.design/book/contents/get-started), Testimonium Ltd., 2023. 	

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

- Cursul respectă curriculumul IEEE și ACM pentru domeniul Informatică.
- Cursul există în programele de studiu ale universităților importante din România și din străinătate.
- Conținutul disciplinei este considerat de majoritatea companiilor software ca fiind deosebit de important pentru obținerea unor abilități avansate de programare.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Prezentare pe durata semestrului	Notarea este bazată pe calitatea prezentării, nivelul de detaliu și adecvarea exemplurilor prezentate.	25%
	Examinare pe durata ultimei săptămâni		50%
10.5 Seminar/laborator	Proiect de laborator		25%
10.6 Standard minim de performanță			
<ul style="list-style-type: none">• Studenții vor respecta standardele de integritate academică.• Nota minimă de promovare este definită prin atingerea a cel puțin 50% (5/10) din punctele aferente notei finale.			

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

Nu se aplică.

Data completării:
28.04.2025

Semnătura titularului de curs
Conf. Univ. Dr. Molnar Arthur-Jozsef

Semnătura titularului de seminar
Conf. Univ. Dr. Molnar Arthur-Jozsef

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.*".