

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Automatică și Calculatoare
1.3 Departamentul	Calculatoare
1.4 Domeniul de studii	Calculatoare și Tehnologia Informației
1.5 Ciclul de studii	Master
1.6 Programul de studii / Calificarea	Complemente de Știința Calculatoarelor / Master
1.7 Forma de învățământ	IF – învățământ cu frecvență
1.8 Codul disciplinei	8.

2. Date despre disciplină

2.1 Denumirea disciplinei	Proiectare software				
2.2 Titularii de curs	Prof. dr. ing. Mihaela Dînsoreanu - Mihaela.Dinsoreanu@cs.utcluj.ro				
2.3 Titularul / Titularii activităților de Seminar / laborator / proiect	Prof. dr. ing. Mihaela Dînsoreanu - Mihaela.Dinsoreanu@cs.utcluj.ro				
2.4 Anul de studiu	I	2.5 Semestrul	2	2.6 Tipul de evaluare (E – examen, C – colocviu, V – verificare)	E
2.7 Regimul disciplinei	DA – de aprofundare, DS – de sinteza, DC – complementară				DA
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

3. Timpul total estimat

3.1 Număr de ore pe săptămână	5	din care:	Curs	2	Seminar		Laborator	2	Proiect	1
3.2 Număr de ore pe semestru	70	din care:	Curs	28	Seminar		Laborator	28	Proiect	14
3.3 Distribuția fondului de timp (ore pe semestru) pentru:										
(a) Studiul după manual, suport de curs, bibliografie și notițe										25
(b) Documentare suplimentară în bibliotecă, pe platforme electronice de specialitate și pe teren										25
(c) Pregătire seminarii / laboratoare, teme, referate, portofolii și eseuri										20
(d) Tutoriat										4
(e) Examinări										6
(f) Alte activități:										0
3.4 Total ore studiu individual (suma (3.3(a)...3.3(f)))										80
3.5 Total ore pe semestru (3.2+3.4)										150
3.6 Numărul de credite										6

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Tehnici de programare
4.2 de competențe	Operarea cu fundamente științifice, ingineresti și ale informaticii

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Tabla, proiector, calculator, Moodle, Webex
5.2. de desfășurare a seminarului / laboratorului / proiectului	Calculatoare, software specific, GitHub, Skype Prezenta la laborator si proiect este obligatorie

6. Competențele specifice acumulate

6.1 Competențe profesionale	<p>C2 - Elaborarea de tehnici, tehnologii, metode și metodologii specifice sistemelor informatice</p> <ul style="list-style-type: none"> • C2.1 - Recunoașterea de tehnici, metode, metodologii și tehnologii utilizate în sistemele informatice • C2.2 - Fixarea condițiilor de utilizare pentru diferite platforme de calcul din domeniul sistemelor informatice • C2.3 - Dezvoltarea de aplicații bazate pe noi tehnici, metode și metodologii destinate sistemelor informatice • C2.4 - Evaluarea necesarului de tehnologii, resurse, echipamente și integrarea și adaptarea acestora în sisteme informatice
-----------------------------	---

	<ul style="list-style-type: none"> • C2.5 - Cercetarea și dezvoltarea de tehnici, metode și metodologii noi specifice sistemelor informatice <p>C3 - Proiectarea inovativă a sistemelor informatice dedicate</p> <ul style="list-style-type: none"> • C3.1 - Identificarea și descrierea tehnicilor, metodelor, metodologiilor și tehnologiilor necesare în proiectarea sistemelor informatice • C3.2 - Utilizarea de concepte, principii, tehnici, metodologii și tehnologii de proiectare a sistemelor informatice • C3.3 - Crearea și utilizarea de soluții noi adecvate, în realizarea de proiecte de sisteme informatice • C3.4 - Evaluarea efectelor alternativelor de rezolvare în creșterea performanțelor sistemelor informatice • C3.5 - Elaborarea de soluții eficiente în proiectarea sistemelor informatice prin selectarea alternativelor specifice domeniului
6.2 Competențe transversale	N/A

7. Obiectivele disciplinei

7.1 Obiectivul general al disciplinei	Principalul obiectiv al acestei discipline este de a oferi informații specifice și de a pregăti studenții în vederea realizării de proiecte software conforme cu bunele practici. Astfel, se urmărește conferirea capacității de a analiza diferite alternative arhitecturale și de proiectare, de a lua deciziile arhitecturale cele mai potrivite contextului în vederea proiectării oricărui tip de aplicație, cu accent pe optimizarea cât mai pronunțată a performanțelor acestuia.
7.2 Obiectivele specifice	<p>Pentru atingerea acestor obiective generale, studenții vor:</p> <ul style="list-style-type: none"> • Urmări să înțeleagă și să rezolve atât cerințele funcționale ale unui sistem software cât și cerințele de calitate ale acestuia (disponibilitate, performanță, securitate, scalabilitate etc) • Studia soluții arhitecturale existente pe diferite nivele de granularitate (stiluri arhitecturale, șabloane arhitecturale și de proiectare) • Studia principii și paradigme de proiectare a claselor și pachetelor • Învăța să analizeze cerințele și să proiecteze alternative arhitecturale aplicabile; • Învăța să aplice principii și paradigme avansate de proiectare, precum și tehnici descriptive specifice;

8. Conținuturi

8.1 Curs	Nr.ore	Metode de predare	Observații
Introducere. Principii de proiectare a claselor (SOLID)	2	Scenariul onsite : - Mijloace multimedia - Prezentări PowerPoint - Demonstrații pe tablă	
Principii de proiectare a claselor (GRASP) și pachetelor	2		
Sabloane arhitecturale (Layers, Event-driven, MVC)	2		
Domain-driven design	2		
Service-oriented design	2		
Examen parțial/Demonstratie de cod	2		
Sabloane aplicate în arhitecturi Enterprise (Accesul la resurse)	2		
Sabloane aplicate în arhitecturi Enterprise (Prezentare)	2		
Sabloane aplicate în arhitecturi Enterprise (Concurența)	2		
Aplicarea șabloanelor de proiectare (creationale)	2		
Aplicarea șabloanelor de proiectare (structurale)	2		
Aplicarea șabloanelor de proiectare (comportamentale)	2		
Atribute de calitate și Metrici ale calitatii proiectelor software	2		
Recapitulare și pregătire pentru examen	2		
Bibliografie (<i>bibliografia minimală a disciplinei conținând cel puțin o lucrare bibliografică de referință a disciplinei, care există la dispoziția studenților într-un număr de exemplare corespunzător</i>)			
1. Juval Lowy, Righting software, O'Reilly, 2020 2. Mark Richards, Software Architecture Patterns, O'Reilly, 2015 3. Vaughn Vernon, Domain Driven Design Distilled, Addison Wesley, 2016 4. Ian Gorton, Essential Software Architecture, Springer, second ed. 2011.			

5. Taylor, R., Medvidovic, N., Dashofy, E., Software Architecture: Foundations, Theory, and Practice, 2010, Wiley.
6. Len Bass, Paul Clements, Rick Kazman, Software Architecture in Practice, 3rd edition, 2013.
7. Buschmann, Frank, Regine Meunier, Hans Rohnert, Peter Sornmerlad, and Michael Stal. 2001. *Pattern-oriented system architecture, volume 1: A system of patterns*. Hoboken, NJ: John Wiley & Sons.
8. Fowler Martin, *Patterns of Enterprise Application Architecture*, Addison-Wesley Professional, 2002
9. E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Design Patterns. AddisonWesley, 1995.
10. Craig Larman, *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development* (3rd Edition), Prentice Hall, 2004, ISBN: 0131489062

Materialele de curs sunt publicate pe moodle.cs.utcluj.ro

8.2 Aplicații (seminar/laborator/proiect)*	Nr.ore	Metode de predare	Observații
Exercitii recapitulative (UML, principii de proiectare clase si pachete)	3	Scenariul onsite : Implementarea temelor pe sistemele din laborator, prezentari si demonstratii la tabla	
Conectare la baze de date si operatii pe baze de date	3		
Stiluri arhitecturale	3		
Stiluri arhitecturale	3		
Prezentare tema 1	3		
Sabloane arhitecturale	3		
Sabloane arhitecturale	3		
Prezentare tema 2	3		
Sabloane de proiectare (creationale)	3		
Sabloane de proiectare (structurale)	3		
Sabloane de proiectare (comportamentale)	3		
Prezentare tema 3	3		
Recuperari	3		
Recuperari	3		

Bibliografie (*bibliografia minimală pentru aplicații conținând cel puțin o lucrare bibliografică de referință a disciplinei care există la dispoziția studenților într-un număr de exemplare corespunzător*)

- 1. Bibliografia de curs, materiale de curs publicate la moodle.cs.utcluj.ro
- Tutorialul de Java – docs.oracle.com
- Tutorial de C# - msdn.microsoft.com

*Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.

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

Întrucât această disciplină este foarte importantă pentru proiectarea sistemelor software, conținutul ei este cât se poate de modern deoarece recapitulează principiile, apoi soluțiile arhitecturale consacrate și în final prezintă metrici de evaluare a proiectelor software. Conținutul disciplinei a fost discutat cu actori importanți din acest domeniu, atât academici cât și industriali, din România, Europa și S.U.A. Disciplina a fost evaluată, o dată cu programul de studiu de master Complemente de Știința Calculatoarelor, de către ARACIS.

10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	Abilitatea de rezolvare a unor probleme specifice domeniului Prezență, (inter)activitate în timpul orelor de curs	Examen scris	60%
Seminar	-		
Laborator	Abilitatea de rezolvare a unor probleme specifice domeniului Prezență, (inter)activitate în timpul orelor de aplicații	Teme de laborator, proiect GitHub	40%
Proiect	-		

Standard minim de performanță:

Analiza cerințelor și proiectarea unei soluții corecte a unei aplicații software, utilizând aparatul formal specific domeniului.

Calcul nota disciplina: 20% laborator + 20% proiect + 60% examen final

Conditii de participare la examenul final: Laborator \geq 5, Proiect \geq 5

Conditii de promovare: Examen final \geq 5

Data	Titulari	Titlu Prenume NUME	Semnătura
completării: 28.06.2023	Curs	Prof. dr. ing. Mihaela Dinsoreanu	
	Aplicații	Prof. dr. ing. Mihaela Dinsoreanu	

Data avizării în Consiliul Departamentului Calculatoare

Director Departament,
Prof. dr. ing. Rodica Potolea

Data aprobării în Consiliul Facultății de Automatică și Calculatoare

Decan,
Prof. dr. ing. Liviu Miclea