

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Babeș-Bolyai Cluj-Napoca
1.2 Facultatea	Matematică și Informatică
1.3 Departamentul	Informatică
1.4 Domeniul de studii	Informatică
1.5 Ciclul de studii	licență
1.6 Programul de studiu / Calificarea	Informatică linia de studiu română

2. Date despre disciplină

2.1 Denumirea disciplinei	Ingineria sistemelor soft						
2.2 Titularul activităților de curs	Lect. dr. Vladiaela Petrașcu						
2.3 Titularul activităților de seminar	Lect. dr. Vladiaela Petrașcu						
2.4 Anul de studiu	2	2.5 Semestrul	4	2.6. Tipul de evaluare	C	2.7 Regimul disciplinei	obligatorie

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

3.1 Număr de ore pe săptămână	5	Din care: 3.2 curs	2	3.3 seminar/laborator	3
3.4 Total ore din planul de învățământ	70	Din care: 3.5 curs	28	3.6 seminar/laborator	42
Distribuția fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie și notițe					14
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					14
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					28
Tutoriat					14
Examinări					10
Alte activități:					
3.7 Total ore studiu individual	80				
3.8 Total ore pe semestru	150				
3.9 Numărul de credite	6				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	<ul style="list-style-type: none"> • Fundamentele programării • Programare orientată obiect
4.2 de competențe	<ul style="list-style-type: none"> • Abilități de programare într-un limbaj orientat-obiect de nivel înalt

5. Condiții (acolo unde este cazul)

5.1 De desfășurare a cursului	<ul style="list-style-type: none"> • Videoproiector
5.2 De desfășurare a seminarului/laboratorului	<ul style="list-style-type: none"> • Calculatoare • Instrument CASE UML • Java / .NET IDE

6. Competențele specifice acumulate

Competențe profesionale	<p>C2.1 Identificarea de metodologii adecvate de dezvoltare a sistemelor software</p> <p>C2.2 Identificarea și explicarea mecanismelor adecvate de specificare a sistemelor software</p> <p>C2.3 Utilizarea metodologiilor, mecanismelor de specificare și a mediilor de dezvoltare pentru realizarea aplicațiilor informatice</p> <p>C2.5 Realizarea unor proiecte informatice dedicate</p>
Competențe transversale	<p>CT1 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</p> <p>CT2 Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar și dezvoltarea capacităților empatiche de comunicare inter-personală, de relaționare și colaborare cu grupuri diverse</p> <p>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</p>

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> • Cunoașterea și aplicarea unor concepte, principii și tehnici ingineresti solide de dezvoltare a sistemelor soft
7.2 Obiectivele specifice	<ul style="list-style-type: none"> • Cunoașterea etapelor ciclului de viață al softului și a modelelor de procese soft • Înțelegerea conceptelor legate de modelarea softului • Cunoașterea și aplicarea tehnicilor de dezvoltare a softului pe baza modelelor • Familiarizarea cu limbajul UML • Abilitatea de a utiliza instrumente CASE • Însușirea aspectelor de bază legate de gestiunea softului • Familiarizarea cu unele dintre metodologiile de dezvoltare, tradiționale sau agile

8. Conținuturi

8.1 Curs	Metode de predare	Observații
1. Introducere în ingineria sistemelor soft: motivație, definiție, concepte, activități	expunerea, conversația, discutarea unor studii de caz	
2. Ciclul de viață al softului. Modele de procese soft (secvențiale, iterative, specializate)	expunerea, conversația, discutarea unor studii de caz	
3. Tehnici de gestionare a complexității softului (abstractizare, descompunere, modelare). Modelarea în ingineria sistemelor soft: definiții, tipuri de modele și instrumente de modelare	expunerea, conversația, discutarea unor studii de caz	
4. Modelarea sistemelor soft folosind UML: concepte de	expunerea, conversația,	

modelare, tipuri de diagrame, notații, instrumente	discutarea unor studii de caz	
5. Colectarea cerințelor: concepte, activități, exemple	expunerea, conversația, discutarea unor studii de caz	
6. Analiza cerințelor: concepte, activități, exemple	expunerea, conversația, discutarea unor studii de caz	
7. Proiectarea sistemelor soft: concepte, principii, activități. Proiectare arhitecturii	expunerea, conversația, discutarea unor studii de caz	
8. Proiectare sistemelor soft. Proiectarea detaliată - șabloane de proiectare	expunerea, conversația, discutarea unor studii de caz	
9. Proiectare sistemelor soft. Proiectarea detaliată - specificarea interfețelor	expunerea, conversația, discutarea unor studii de caz	
10. Implementarea sistemelor soft. Transformarea modelelor în cod: concepte, principii, activități, exemple	expunerea, conversația, discutarea unor studii de caz	
11. Verificarea și validarea sistemelor soft	expunerea, conversația, discutarea unor studii de caz	
12. Gestiunea proiectelor soft: concepte și activități	expunerea, conversația, discutarea unor studii de caz	
13. Metodologii de dezvoltare a sistemelor soft	expunerea, conversația, discutarea unor studii de caz	
14. Evaluare		

Bibliografie

- [1] Booch, G., Rumbaugh, J., Jacobson, I., *The Unified Modeling Language User Guide - V.2.0*, Addison Wesley, 2005.
- [2] Bruegge, B., Dutoit, A., *Object-Oriented Software Engineering Using UML, Patterns and Java - 3rd Edition*, Prentice Hall, 2009.
- [3] Fowler, M. et al., *Refactoring - Improving the Design of Existing Code*, Addison Wesley, 1999.
- [4] Fowler, M., Scott, K., *UML Distilled: A Brief Guide to the Standard Object Modeling Language -3rd ed.*, Addison-Wesley, 2003.
- [5] Gamma, E., Helm, R., Johnson, R., Vlissides, J., *Design Patterns*, Addison-Wesley, 1996.
- [6] Martin, R.C., *Agile Software Development: Principles, Patterns, and Practices*, Prentice Hall, 2002.
- [7] Pârv, B., *Analiza și proiectarea sistemelor*, Univ. Babeș-Bolyai, CFCID, Facultatea de Matematică și Informatică, Cluj-Napoca, 2004.
- [8] Pressman, R.S., *Software Engineering - A Practitioners Approach - 8th ed.*, McGraw-Hill, 2014.
- [9] Rumbaugh, J., Jacobson, I., Booch, G., *The Unified Modeling Language Reference Manual (2nd edition)*, Addison Wesley, 2010.
- [10] Schach, S.R., *Object-Oriented and Classical Software Engineering - 8th ed.*, McGraw-Hill, 2010.
- [11] Sommerville, I., *Software Engineering - 10th edition*, Addison-Wesley, 2015.

8.2 Seminar	Metode de predare	Observații
1. Modelul functional al unui sistem. Diagrame UML de cazuri de utilizare: concepte, relații, reprezentare, structura documentului de descriere a cazurilor de utilizare	explicația, dialogul, exemplificarea	Un seminar de 2 ore la două săptămâni
2. Modele structurale ale unui sistem. Diagrame UML de clase: concepte, relații, reprezentare. Modelul conceptual	explicația, dialogul, exemplificarea	
3. Modele comportamentale ale unui sistem. Diagrame UML de secvență: utilizare, concepte, reprezentare. Modelul obiectual de proiectare	explicația, dialogul, exemplificarea	
4. Diagrame UML de comunicare: concepte, reprezentare, echivalenta cu diagramele de secvență	explicația, dialogul, exemplificarea	
5. Diagrame UML de tranziție a stărilor: concepte, reprezentare. Sablonul de proiectare <i>State</i>	explicația, dialogul, exemplificarea	
6. Rolul asertiunilor în specificarea modelelor. Design by	explicația, dialogul,	

Contract. Limbajul OCL	exemplificarea	
7. Generarea codului pe baza diagramelor UML. Ingineria software dirijata de modele (<i>model-driven engineering, MDE</i>)	explicația, dialogul, exemplificarea	
<p>Bibliografie</p> <p>[1] Object Management Group, UML 2.4.1 Infrastructure Specification - 2012 ISO standard, https://www.omg.org/spec/UML/ISO/19505-1/PDF.</p> <p>[2] Object Management Group, UML 2.4.1 Superstructure Specification - 2012 ISO standard, https://www.omg.org/spec/UML/ISO/19505-2/PDF.</p> <p>[3] Object Management Group, OCL 2.3.1 Specification - 2012 ISO standard , https://www.omg.org/spec/OCL/ISO/19507/PDF.</p> <p>[4] Object Management Group, UML 2.5.1 Specification, 2017, https://www.omg.org/spec/UML/2.5.1/PDF.</p> <p>[5] Object Management Group, OCL 2.4 Specification, 2014, https://www.omg.org/spec/OCL/2.4/PDF.</p>		
8.3 Laborator	Metode de predare	Observații
1. Metodologii soft agile - planificarea etapelor de dezvoltare a softului. Atribuirea fiecărui student a unei aplicații de mici dimensiuni, pe care acesta va trebui sa o analizeze, proiecteze, implementeze și testeze.	problematizarea, exemplificarea	Un laborator de 2 ore la două săptămâni
2. Utilizarea unui instrument CASE UML și a unui editor de texte pentru realizarea modelului funcțional al aplicației	problematizarea, exemplificarea	
3. Utilizarea unui instrument CASE UML pentru realizarea modelului conceptual	problematizarea, exemplificarea	
4. Utilizarea unui instrument CASE UML pentru realizarea modelului de proiectare	problematizarea, exemplificarea	
5. Utilizarea unui instrument CASE UML și a unui IDE pentru implementarea aplicației	problematizarea, exemplificarea	
6. Testarea aplicației realizate	problematizarea, exemplificarea	
7. Elaborarea manualului de utilizare și predarea aplicației și a documentației aferente	problematizarea, exemplificarea	
<p>Bibliografie</p> <p>[1] Kroll, P., Kruchten P., and Booch, G., <i>The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP</i>, Addison-Wesley, 2003.</p> <p>[2] Kruchten, P., <i>The Rational Unified Process: An Introduction - 3rd Edition</i>, Addison-Wesley, 2003.</p> <p>[3] Rubin, K.S., <i>Essential Scrum - A Practical Guide to the Most Popular Agile Process</i>, Addison-Wesley 2012.</p>		

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

- Cursul respectă recomandările curriculare IEEE / ACM pentru programele de studii de informatică.
- Cursuri cu conținut similar sunt predare la majoritatea universităților din România care au programe de studii similare.
- Companiile de dezvoltare de software consideră foarte important conținutul cursului pentru formarea viitorilor dezvoltatori de software.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	<ul style="list-style-type: none">• Cunoașterea conceptelor și principiilor de bază ale ingineriei sistemelor soft predate la curs• Insușirea conceptelor legate de modelarea softului, precum și abilitatea de a utiliza limbajul UML în acest scop	Examen scris	60%
10.5 Seminar/laborator	<ul style="list-style-type: none">• Aplicarea cunoștințelor acumulate la curs pentru rezolvarea unor probleme concrete	Proiect de laborator Activitate în timpul seminariilor	40%
10.6 Standard minim de performanță			
<ul style="list-style-type: none">• minim nota 5, atât la examenul scris, cât și la evaluarea activității de seminar/laborator			

Data completării

22 aprilie 2019

Semnătura titularului de curs

Lect. dr. Vladiea PETRAȘCU

Semnătura titularului de seminar

Lect. dr. Vladiea PETRAȘCU

Data avizării în departament

.....

Semnătura directorului de departament

Prof. dr. Anca ANDREICA