

FI A DISCIPLINEI

1. Date despre program

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

2. Date despre disciplină

2.1 Denumirea disciplinei	Ingineria sistemelor soft						
2.2 Titularul activităților de curs	conf. dr. Dan CHIOREAN						
2.3 Titularul activităților de seminar	conf. dr. Dan CHIOREAN						
2.4 Anul de studiu	3	2.5 Semestrul	6	2.6. Tipul de evaluare	E	2.7 Regimul disciplinei	Obligatoriu

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

3.1 Număr de ore pe săptămână	4	Din care: 3.2 curs	2	3.3 seminar/laborator	2
3.4 Total ore din planul de învățământ	48	Din care: 3.5 curs	24	3.6 seminar/laborator	24
Distribuția fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie și notițe					22
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					18
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					30
Tutoriat					28
Examinări					4
Alte activități:					
3.7 Total ore studiu individual					102
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"> Programare Orientată -Obiect
4.2 de competențe	<ul style="list-style-type: none"> cunoștințe medii de programare într-un limbaj de programare OO: Java, C# sau C++

5. Condiții (acolo unde este cazul)

5.1 De desfășurare a cursului	<ul style="list-style-type: none"> video-proiector
5.2 De desfășurare a seminarului/laboratorului	<ul style="list-style-type: none"> Laboratoare dotate cu calculatoare care să aibă instalat JVM, un mediu de dezvoltare: Eclipse IDE, Visual Studio, IntelliJ IDEA

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>C2.4 Utilizarea de criterii și metode adecvate pentru evaluarea 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 (reie îndin 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 • Înșuirea aspectelor de bază legate de gestiunea softului • Familiarizarea cu unele dintre metodologiile de dezvoltare, tradiționale sau agile • Înțelegerea conceptelor legate de modelarea softului • Cunoașterea și aplicarea tehnicilor de dezvoltare a softului pe baza modelelor, precum și a diferitor metode de testare • Familiarizarea cu limbajul UML • Abilitatea de a utiliza instrumente CASE

8. Conținut

8.1 Curs	Metode de predare	Observații
1. Introducere în ingineria sistemelor soft: motivație, definiție, concepte, activități	expunere, conversație, discutarea unor studii de caz	
2. Ciclul de viață al softului. Modele de procese soft (secvențiale, iterative, specializate)	expunere, conversație, 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	expunere, conversație, discutarea unor studii de caz	
4. Modelarea sistemelor soft folosind UML: concepte de modelare, tipuri de diagrame, notații, instrumente	expunere, conversație, discutarea unor studii de caz	
5. Colectarea cerințelor: concepte, activități, exemple	expunere, conversație, discutarea unor studii de caz	
6. Analiza cerințelor: concepte, activități,	expunere, conversație,	

exemple	discutarea unor studii de caz	
7. Proiectarea sistemelor soft: concepte, principii, activități. Proiectare arhitecturii	expunere, conversație, discutarea unor studii de caz	
8. Proiectare sistemelor soft. Proiectarea arhitecturii (subsisteme)	expunere, conversație, discutarea unor studii de caz	
9. Proiectarea sistemelor soft. Abloane de proiectare	expunere, conversație, discutarea unor studii de caz	
10. Proiectare sistemelor soft. Specificarea interfelelor	expunere, conversație, discutarea unor studii de caz	
11. Implementarea sistemelor soft. Transformarea modelelor în cod: concepte, principii, activități, exemple	expunere, conversație, discutarea unor studii de caz	
12. Verificarea și validarea sistemelor soft	expunere, conversație, discutarea unor studii de caz	

Bibliografie:

- Booch, G., Rumbaugh, J., Jacobson, I., *The Unified Modeling Language User Guide - V.2.0*, Addison Wesley, 2005.
- Wesley, 2005.
- Bruegge, B., Dutoit, A., *Object-Oriented Software Engineering Using UML, Patterns and Java - 3rd Edition*, Prentice Hall, 2009.
- Fowler, M. et al., *Refactoring - Improving the Design of Existing Code*, Addison Wesley, 1999.
- [owler, M., Scott, K., *UML Distilled: A Brief Guide to the Standard Object Modeling Language -2nd ed.*, Addison-Wesley, 1999.
- Gamma, E., Helm, R., Johnson, R., Vlissides, J., *Design Patterns*, Addison-Wesley, 1996.
- Martin, R.C., *Agile Software Development: Principles, Patterns, and Practices*, Prentice Hall, 2002.
- Pârv, B., *Analiza și proiectarea sistemelor*, Univ. Babe -Bolyai, CFCID, Facultatea de Matematică și Informatică, Cluj-Napoca, 2004.
- Pressman, R.S., *Software Engineering - A Practitioners Approach - 6th ed.*, McGraw-Hill, 2005.
- Schach, S.R., *Object-Oriented and Classical Software Engineering - 6th ed.*, McGraw-Hill, 2005.
- Sommerville, I., *Software Engineering - 8th edition*, Addison-Wesley, 2006.

8.2 Seminar	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 să o analizeze, proiecteze, implementeze și testeze.	problematizare și exemplificare	Laborstoarele se desfășoară o dată la două săptămâni, 2 ore.
2. Utilizarea unui instrument CASE UML și a unui editor de texte pentru realizarea modelului funcțional al aplicației	problematizare și exemplificare	
3. Utilizarea unui instrument CASE UML pentru realizarea modelului conceptual	problematizare și exemplificare	
4. Utilizarea unui instrument CASE UML pentru realizarea modelului de proiectare	problematizare și exemplificare	
5. Utilizarea unui instrument CASE UML și a unui IDE pentru implementarea aplicației	problematizare și exemplificare	
6. Testarea, elaborarea manualului de utilizare și predarea aplicației și a documentației aferente	problematizare și exemplificare	

Bibliografie:

- Kroll, P., Kruchten P., and Booch, G., *The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*, Addison-Wesley, 2003.
- Kruchten, P., *The Rational Unified Process: An Introduction - 3rd Edition*, Addison-Wesley, 2003.
- Rubin, K.S., *Essential Scrum - A Practical Guide to the Most Popular Agile Process*, Addison-Wesley 2012.

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

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

10. Evaluare

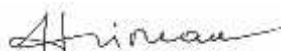
Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	<ul style="list-style-type: none">• cunoașterea conceptelor de bază și a principiilor Ingineriei Programării predate la curs	<ul style="list-style-type: none">• Examen scris	60%
10.5 Seminar/laborator	Aplicarea cunoștințelor acumulate la curs pentru rezolvarea unor probleme concrete	Proiect de laborator Activitate în timpul seminariilor	40%

Data completării

20 aprilie 2018

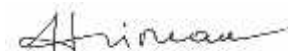
Titular de curs

conf. dr. Dan CHIOREAN



Titular de seminar

conf. dr. Dan CHIOREAN



Data avizării în departament

Director de departament

prof. dr. Anca ANDREICA