

**FIȘA DISCIPLINEI
INGINERIE SOFTWARE**

1. Date despre program

1.1. Instituția de învățământ	Universitatea „1 Decembrie 1918”
1.2. Facultatea	de Științe
1.3. Departamentul	de Științe Exacte și Inginerești
1.4. Domeniul de studii	Științe exacte
1.5. Ciclul de studii	Masterat
1.6. Programul de studii	Programare avansata si baze de date

2. Date despre disciplină

2.1. Denumirea disciplinei	Inginerie software		2.2. Cod disciplină	MI101			
2.3. Titularul activității de curs	Conf. Univ. Dr. Ing. Mircea Risteiu						
2.4. Titularul activității de laborator	Lect.univ.dr. Huțanu C-tin						
2.5. Anul de studiu	I	2.6. Semestrul	I	2.7. Tipul de evaluare (E/C/VP)	E	2.8. Regimul disciplinei (O – obligatorie, Op – opțională, F – facultativă)	O

3. Timpul total estimat

3.1. Numar ore pe saptamana	2	din care: 3.2. curs	2	3.3. laborator	0
3.4. Total ore din planul de învățământ	28	din care: 3.5. curs	28	3.6. seminar/laborator	28
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					34
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					-
Tutoriat					-
Examinări					2
Alte activități					-

3.7 Total ore studiu individual	56
3.8 Total ore din planul de învățământ	28
3.9 Total ore pe semestru	84
3.10 Numărul de credite	3

4. Precondiții

4.1. de curriculum	Programarea calculatoarelor
4.2. de competențe	C1 Utilizarea elementelor fundamentale referitoare la dispozitivele, circuitele, sistemele, instrumentația și tehnologia electronică C3.4 Elaborarea de programe într-un limbaj de programare general și/sau specific, pornind de la specificarea cerințelor și până la execuție, depanare și interpretarea rezultatelor în corelație cu procesorul utilizat

5. Condiții

5.1. de desfășurare a cursului	Videoproiector, Laptop, Copiator, Woofers și cursuri multimedia
5.2. de desfășurarea a seminarului/laboratorului	Sala dotata cu videoproiector/tabla, standuri de laborator specifice, rețea de calculatoare conectata la internet

6. Competențe specifice acumulate

Competențe profesionale	C4.1 Definirea conceptelor, principiilor și metodelor folosite în domeniile: programarea calculatoarelor, limbaje de nivel înalt și specifice, tehnici CAD de realizare a modulelor electronice, microcontrolere, arhitectura sistemelor de calcul, sisteme electronice programabile, grafică,
-------------------------	--

	<p>arhitecturi hardware reconfigurabile</p> <p>C3.3 Rezolvarea problemelor practice concrete care includ elemente de structuri de date și algoritmi, programare și utilizare de microprocesoare sau microcontrolere</p> <p>C4.4 Utilizarea criteriilor de performanță adecvate pentru evaluarea, inclusiv prin simulare, a hardware-ului și software-ului unor sisteme dedicate sau a unor activități de servicii în care se folosesc microcontrolere sau sisteme de calcul de complexitate redusă sau medie</p> <p>C3.5 Realizarea de proiecte care implică componente hardware (procesoare) și software (programare)</p>
Competențe transversale	

7. Obiectivele disciplinei

7.1 Obiectivul general al disciplinei	<p>La terminarea cursului, studenții trebuie să posede:</p> <p>a) cunoștințe sistematice despre metodologiile de realizare a produselor soft</p> <p>b) familiarizarea cu conceptele și preocupările moderne în scrierea softului de aplicație</p> <p>c) cunoștințe despre tipurile de modele și instrumentele de modelare folosite în dezvoltarea aplicațiilor soft</p> <p>d) motivația necesară privind importanța documentării produselor soft, în toate fazele ciclului de viață</p>
7.2 Obiectivele specifice	<p>Să știe să utilizeze mediile de programare în scopul realizării de produse software;</p> <p>Să știe să programeze în limbaj de asamblare și în limbaje OO;</p> <p>Să știe să folosească tehnici de modelarea conceptuală a datelor;</p> <p>Să știe să facă studii de caz complete;</p> <p>Să știe să interpreteze rezultatele obținute.</p>

8. Conținuturi

8.1 Curs	Metode de predare	Observații
Curs 1 – Procese soft: Ciclul de viață al unui program. Ingineria programării: evoluție, definiții. Prototipizarea. Programarea extremă (Extreme Programming, XP);	Prelegere, discutii	
Curs 2 – Dezvoltarea sistemelor soft: principii și activități. Problem solving și dezvoltarea de soft: asemănări și deosebiri. Comunicarea: principii și sarcini generice. Planificarea: principii și sarcini generice. Modelarea în analiză: principii și sarcini generice. Modelarea în proiectare: principii și sarcini generice. Construcția: principii și sarcini generice. Exploatarea: principii și sarcini generice;	Prelegere, discutii	
Curs 3 – Modelarea proceselor și sistemelor soft. Domeniul aplicației: procese și produse. Instrumente folosite în modelarea sistemelor soft;	Prelegere, discutii	
Curs 4 – Limbajul unificat de modelare (UML). Prezentare generală. Modelarea cerințelor. Modelarea claselor. Modelarea dinamică. Modelarea arhitecturii;	Prelegere, discutii	
Curs 5 – Colectarea, analiza și specificarea cerințelor. Procesul de analiză a cerințelor. Inițierea analizei cerințelor. Tehnici de comunicare și de stabilire a cerințelor. Structurarea cerințelor. Specificarea cerințelor. Validarea cerințelor;	Prelegere, discutii	
Curs 6 – Modelarea în analiză. Definiții, metode, elemente, etape. Modelarea cazurilor de utilizare. Modelarea statică. Modelarea dinamică. Modelarea fluxurilor;	Prelegere, discutii	
Curs 7 – Proiectarea softului. Modele și concepte. Fundamentele proiectării. Pașii proiectării. Concepte de proiectare. Modele de proiectare.	Prelegere, discutii	
Curs 8 – Proiectarea arhitecturii și claselor. Principii. Arhitectură și dependențe. Principii de proiectare a claselor. Principii de proiectare a arhitecturii (reutilizarea modulelor, cuplarea modulelor). Șabloane de proiectare arhitecturale;	Prelegere, discutii	
Curs 9 – Proiectarea arhitecturii, datelor și prelucrărilor. Proiectarea arhitecturii. Proiectarea datelor. Proiectarea prelucrărilor;	Prelegere, discutii	
Curs 10 – Proiectarea componentelor. Arhitectura logică a sistemului. Proiectarea dirijată de responsabilități. Pașii proiectării OO;	Prelegere, discutii	
Curs 11 – Proiectarea interfeței cu utilizatorul. Sarcinile proiectării interfeței cu utilizatorul. Prelucrare manuală, batch și on-line. Proiectarea documentelor de intrare și a rapoartelor. Interacțiunea om-calculator;	Prelegere, discutii	
Curs 12 – Testarea softului. Etape și metode. Verificarea și validarea. Testarea softului:	Prelegere, discutii	

etape. Metode de testare;		
Curs 13 – Impachetarea și instalarea. Punerea în exploatare și întreținerea;	Prelegere, discutii	
Curs 14 – Planificarea proiectelor soft. Factorii umani implicați în dezvoltarea aplicațiilor soft. Organizarea procesului de realizare a aplicației soft. Alte activități de planificare. Planul proiectului soft;	Prelegere, discutii	

8.2 Bibliografie

1. Flaaten, P.O., McCubbrey, D.J., O’Riordan, P.D., Burgess, K.: Foundations of Business Systems, Dryden Press, 1st ed. 1989, 2nd ed. 2013.
2. Fowler, M., Scott, K.: UML Distilled: A Brief Guide to the Standard Object Modeling Language, Addison-Wesley, 2nd ed., 2012.
3. Iacobson, I., Booch, G., Rumbaugh, J.: The Unified Software Development Process, Addison-Wesley, 2012.
4. Martin, R.C.: Agile Software Development: Principles, Patterns, and Practices, Prentice Hall, 2012.
5. Pressman, R.S.: Software Engineering - A Practitioners Approach, McGraw-Hill, 3rd ed. 1992; 4th ed. 2006, 5th ed. 2001, 6th ed. 2009.
6. Schach, S.R.: Object-Oriented and Classical Software Engineering, McGraw-Hill, 5th ed., 2002, 6th ed. 2005.
7. Sommerville, I.: Software Engineering, Addison-Wesley, 5th ed. 1996, 6th ed. 2000, 7th ed. 2009.
8. Agile Modeling Homepage, [<http://www.agilemodeling.com>].
9. Software Engineering Body of Knowledge, IEEE, 2004. [<http://www.swebok.org>].

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

- *Elaborata pe baza consultării fișelor disciplinei similare din Centrele universitare Cluj- Napoca (UBB, UT), Timisoara (UP), Bucuresti (UP), Craiova*
- *Pe baza recomandarilor comisiei de evaluare a disciplinei*
- *Feedback din partea studentilor*

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	<i>Evaluare finala</i>	<i>Examen scris, Portofoliu de lucrari practice</i>	100%
10.6 Standard minim de performanță:			
Conceperea unui plan de documentare si de realizarea a unuie aplicatii hardware/ software. Similar oricarui sistem de management al proiectelor, se urmareste obtinerea unui plan de munca, cu etape, chei de verificare, cu plan de indeplinire a cerintelor hardware/ software formulate.			

Data completării

29.09.2016

Semnătura titularului de curs

.....

Semnătura titularului de seminar

.....

Data avizării în departament

3.10.2016

Semnătura director de departament

.....