

INGIENRIA PROGRAMĂRII

1. Date despre unitatea de curs/modul

Facultatea	Calculatoare, Informatică și Microelectronică				
Catedra/departamentul	Informatică și Ingineria Sistemelor				
Ciclul de studii	Studii superioare de licență, ciclul I				
Programul de studiu	0612.1 Calculatoare și rețele				
Anul de studiu	Semestrul	Tip de evaluare	Categoria formativă	Categoria de opționalitate	Credite ECTS
IV (învățământ cu frecvență); V (învățământ cu frecvență redusă)	7 9	E	S – unitate de curs de specialitate	O - unitate de curs obligatorie	4

1. Timpul total estimat

Total ore în planul de învățământ	Din care				
	Ore auditoriale		Lucrul individual		
	Curs	Laborator/seminar	Proiect de an	Studiul materialului teoretic	Pregătire aplicații
Zi 120	30	30/-	-	30	30
FR 120	12	12/-	-	48	48

2. Precondiții de acces la unitatea de curs/modul

Conform planului de învățământ	Pentru a atinge obiectivele cursului studenții trebuie să posede competențe și abilități formate la următoarele unități de curs, prevăzute de planul de învățământ: „Programarea calculatoarelor”, „Tehnici avansate de programare”, „Structuri de date și algoritmi”, „Programarea orientată pe obiecte”, „Testarea sistemelor”, „Programarea concurentă și distribuită”, „Programarea de sistem și de rețea”.
Conform competențelor	Abilități de colaborare și comunicare eficientă în cadrul proiectelor de grup, disciplina implică deseori colaborarea între diferite echipe interdisciplinare, să cunoască limbaje de programare de nivel înalt.

3. Condiții de desfășurare a procesului educațional pentru

Curs	Pentru prezentarea materialului teoretic în sala de curs este nevoie de tablă, proiector și calculator. Nu vor fi tolerate întârzierile studenților, precum și convorbirile telefonice în timpul cursului.
Laborator/seminar	Studenții vor perfecta rapoarte conform condițiilor impuse de indicațiile metodice. Termenul de predare a lucrării de laborator – 2 săptămâni după finalizarea acesteia. Pentru predarea cu întârziere a lucrării aceasta se depunțează cu 1pct./săptămână de întârziere.

4. Competențe specifice acumulate

Competențe profesionale	<p>CPL 3. Integrarea componentelor:</p> <p>K1 Componente/module hardware/software, indiferent dacă sunt vechi, existente sau noi.</p> <p>K2 Impactul integrării unui sistem asupra organizației sau a sistemului existent.</p> <p>K3 Tehnici de interfațare între module, sisteme și componente.</p> <p>K4 Tehnici de testare a integrării.</p> <p>K5 Instrumentele de dezvoltare (ex. mediul de dezvoltare, gestionare, control al modificărilor și accesul la codul sursă).</p> <p>K6 Bune practici de design.</p> <p>CPL 4. Testarea aplicațiilor</p> <p>K1 Tehnicile, infrastructura și instrumentele necesare utilizate în procesul de testare.</p>
-------------------------	--

	<p>K2 Ciclul de viață al unui proces de testare. K3 Tipurile de teste (funcțional, de integrare, performanță, utilizabilitate, sarcină etc.). K4 Standardele naționale și internaționale care definesc criteriile de calitate pentru testare. K5 Specificul tehnologiilor legate de web, cloud, instrumente mobile și de probleme de mediu.</p> <p>CPL 5. Implementarea soluțiilor: K1 Tehnici de analiză a performanței. K2 Tehnicile legate de gestionarea problemelor (funcționare, performanță, compatibilitate). K3 Software-ul de ambalare/packaging și metode și tehnici de distribuție/desfășurare. K4 Impactul implementării/ desfășurării asupra arhitecturii existente. K5 Tehnologiile și standardele care se utilizează în timpul implementării/ /desfășurării.</p> <p>CPL 6. Elaborarea documentației K1 Instrumente pentru producerea, editarea și distribuirea documentelor profesionale. K2 Instrumente pentru crearea de prezentări multimedia. K3 Diferitele documente tehnice necesare pentru proiectarea, dezvoltarea și implementarea produselor, aplicațiilor și serviciilor. K4 Mijloace de gestiune a versiunilor pentru controlul producției de documente.</p> <p>CPL 7. Suportul/Asistența tehnică a utilizatorilor K1 Principalele aplicații informatice ale utilizatorilor. K2 Schemele bazelor de date și organizarea conținutului. K3 Procedurile interne de raportare a incidentelor în cadrul companiei. K4 Metodele și procedurile de distribuție software pentru transmiterea și implementarea de patch-uri de aplicație cu fișierele afectate.</p> <p>CPL 9. Manage men-tul probleme lor K1 Infrastructura IT globală a organizațiilor și componentele-cheie ale acestora. K2 Procedurile pentru raportarea informațiilor din partea organizațiilor. K3 Procedurile pentru raportarea situațiilor critice ale organizațiilor. K4 Domeniul de aplicare și disponibilitatea instrumentelor de diagnosticare. K5 Legătura dintre elementele de infrastructură a sistemului și impactul eșecurilor asupra proceselor relevante ale afacerii.</p> <p>CPL 10. Gestionarea securității informațiilor K1 Politica organizației privind gestionarea securității și implicațiile sale în angajarea față de clienți, furnizori și subcontractanți. K2 Cele mai bune practici și standarde în managementul securității informațiilor. K3 Riscurile critice pentru managementul securității informațiilor. K4 Abordarea auditului intern al TIC. K5 Tehnicile de detectare a securității, inclusiv cele mobile și digitale. K6 Tehnici de atac cibernetic și contra-măsuri pentru evitarea lor. K7 Investigațiile informatice deja realizate</p>
Competențe transversale	<p>CTL1. Autonomie și responsabilitate CTL2. Interacțiune socială CTL3. Dezvoltare personală și profesională</p>

5. Obiectivele unității de curs/modulului

Obiectivul general	<p>Studierea teoretică și practică a metodelor de proiectare și implementare ale produselor software ce înglobează tehnici de fiabilizare, aprofundarea cunoștințelor legate de problemele proiectării diverselor sisteme de prelucrare a informației ca o posibilitate de menținere a fiabilității și disponibilității sistemelor de calcul, studierea și elaborarea testelor pentru verificarea și validarea produselor program</p>
--------------------	---

Obiectivele specifice	<p>Înțelegerea metodologiilor de dezvoltare a programelor.</p> <ul style="list-style-type: none"> • Capacitatea de extragere, identificare și specificare a cerințelor. • Capacitatea de proiectare orientată obiect a programelor. • Cunoștințe în identificarea etapelor de proiectare. • Cunoștințe în modelele de implementare a programelor. • Cunoștințe și abilități în utilizarea limbajului UML pentru modelarea conceptuală a proiectului. • Cunoștințe în verificarea și validarea programelor, tipurile de testare. • Cunoștințe în managementul proiectului, estimarea costurilor, îmbunătățirea procesului de dezvoltare.
-----------------------	--

6. Conținutul unității de curs/modulului

Tematica activităților didactice	Numărul de ore	
	Înv. cu frecvență	Înv. cu fr. redusă
Tematica prelegerilor		
T1. Introducere. Obiectul de studiu al Ingineriei Programării. Modelele principale ale Ingineria Programării. Importanța Ingineriei Programării în pregătirea inginerilor. Definiția sistemelor socio-tehnice și sistemelor critice. Diferența, rolul și importanța lor în societate	2	1
T2. Procesele de dezvoltare. Managementul proiectului. Modelele de dezvoltare a programelor, iterațiile și activitățile. Noțiuni RUP (Rational Unified Process) și CASE (Computer-Aided Software Engineering).	2	1
T3. Cerințele programului. Procesele de inginerie a cerințelor. Cerințe funcționale și ne-funcționale. Cerințele utilizatorului. Cerințele sistemului. Specificații pentru Interfața utilizator.	2	1
T4. Modelele sistemului. Specificațiile sistemelor critice. Specificațiile formale. Modele de comportament. Modele de prezentare a datelor. Modele de prezentare a obiectelor.	2	1
T5. Conceptul arhitectural. Arhitectura sistemelor distribuite. Arhitectura aplicației.	2	1
T6. Conceptul orientat obiect.	2	1
T.7 Reutilizarea soft-ului	2	1
T.8 Implementarea sistemelor critice	2	1
T.9 Verificarea și validarea.	2	1
T.10 Îmbunătățirea procesului.	4	1
T.11 Ingineria securității.	4	1
T.12 Ingineria produselor program orientată pe servicii și pe aspecte	4	1
Total prelegeri:	30	12
Tematica lucrărilor de laborator		
LP1 Elemente ale programului Enterprise Architect. Diagrame de prezentare a concepției proiectului. Repartizarea temelor pentru lucrări de laborator.	8	2
LP2. Extragerea și analiza cerințelor conform temei personale. Crearea documentului de cerințe. Construirea diagramelor Use-Case în baza cerințelor colectate.	8	2
LP3. Estimarea costurilor proiectului. Testarea programelor. Planificarea verificării și validării. Inspecția programului. Verificarea automată. Testarea sistemului. Testarea componentelor. Asigurarea siguranței și securității.	8	4
LP4. Implementarea componentelor sistemului analizat.	6	4

Total lucrări de laborator:	30	12
------------------------------------	-----------	-----------

7. Referințe bibliografice

Principale	<ol style="list-style-type: none"> 1. Ian Sommerville Software Engineering, Eight Edition. Addison-Wesley 2007. 2. Ian Sommerville Software Engineering, Seventh Edition. Addison-Wesley 2004. 3. Ian Sommerville Software Engineering, Sixth Edition. Addison-Wesley 2001. 4. Craig Larman, Applying UML and Patterns, Third Edition. Addison Wesley, October 2004. 5. Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young, Ph.D., Jim Conallen, Kelli A. Houston. Object-Oriented Analysis and Design with Applications, Third Edition. Addison-Wesley, April 30, 2007.
------------	--

3. Evaluare

Forma de învățământ	Periodică		Curentă	Lucrul individual	Examen final
	Atestarea 1	Atestarea 2			
Cu frecvență	15%	15%	15%	15%	40%
Cu frecvență redusă	25%			25%	50%
Standard minim de performanță					
Prezența și activitatea la prelegeri și lucrări de laborator					
Obținerea notei minime de „5” la fiecare dintre evaluări și lucrări de laborator					