

ARHITECTURA SISTEMELOR SOFTWARE
1. Date despre disciplină/modul

Facultatea	Calculatoare, Informatică și Microelectronică				
Departamentul	Inginerie Software și Automatică				
Ciclul de studii	Ciclul II, Studii superioare de master				
Programul de studii	Ingineria software				
Anul de studii	Semestrul	Tip de evaluare	Categoria formativă	Categoria de opționalitate	Credite ECTS
Anul I (<i>învățământ cu frecvență</i>)	I	E	F – uniate de curs fundamentală	O - unitate de curs obligatorie	5

2. Timpul total estimat

Total ore în planul de învățământ	Din care				
	Ore auditoriale		Lucrul individual		
	Curs	Practice	Proiect de an	Studiul materialului teoretic	Pregătire aplicații
Învățământ cu frecvență	20	20	-	110	-

3. Precondiții de acces la disciplină/modul

Conform planului de învățământ	Pentru a atinge obiectivele cursului masteranzii trebuie să posede cunoștințe de bază în programarea calculatoarelor, înțelegerea conceptelor de bază ale arhitecturii calculatoarelor și structurilor de date. Să dețină experiență în utilizarea limbajelor de programare pentru dezvoltarea aplicațiilor software.
Conform competențelor	Abilități de analiză și sinteză a cerințelor unui sistem software. Capacitatea de a lucra în echipă și de a colabora pentru realizarea proiectelor software. Cunoștințe privind tehnologiile software moderne utilizate în dezvoltarea aplicațiilor.

4. Condiții de desfășurare a procesului educațional pentru

Curs	Pentru prezentarea materialului teoretic în sala de curs este nevoie de proiector, PC/laptop și acces la internet.
Practice	Pentru petrecerea lucrărilor practice/seminare în sală este nevoie de tablă, cretă, proiector. Acces la internet, la laptopuri și medii de dezvoltare software.

5. Competențe specifice acumulate

Competențe profesionale	C1. Identificarea și definirea conceptelor, metodelor și practicilor utilizate în proiectarea arhitecturilor software. C2. Explicarea atributelor de calitate și a tehnologiilor potrivite pentru realizarea arhitecturilor software complexe. C3. Aplicarea principiilor arhitecturale moderne pentru a răspunde cerințelor sistemelor software distribuite. C4. Înțelegerea etapelor și tehnicilor de proiectare a arhitecturilor. C5. Formarea capacității de analiză și specificare a cerințelor arhitecturale. C6. Integrarea atributelor de calitate (ex. securitate, modifiabilitate, performanță) în proiectarea arhitecturii. C7. Compararea arhitecturilor software pentru a determina soluția optimă în funcție de cerințele specifice ale proiectului.
Competențe transversale	CT1. Dezvoltarea capacităților de colaborare și comunicare, asumând roluri de conducere sau execuție în echipele de proiectare. CT2. Îmbunătățirea continuă a cunoștințelor profesionale și utilizarea lor pentru soluționarea problemelor complexe din domeniul arhitecturilor software.

6. Obiectivele disciplinei/modulului

Obiectivul general	Dezvoltarea cunoștințelor și competențelor necesare pentru analiza, proiectarea, implementarea și evaluarea arhitecturilor software, cu accent pe integrarea atributelor de calitate în soluțiile software distribuite și complexe.
Obiectivele specifice	- Familiarizarea cu tipurile de arhitecturi (Monolit, SOA, Microservicii, Event-Driven) și atributele de calitate esențiale (disponibilitate, securitate, performanță etc.). - Aplicarea metodelor și instrumentelor pentru proiectarea arhitecturilor în funcție de cerințele unui sistem software. - Realizarea unui proiect complet folosind una dintre arhitecturile studiate, incluzând schema, implementarea și caracterizarea acesteia. - Analiza comparativă a arhitecturii implementate cu alte tipuri de arhitecturi, având în vedere avantajele și limitările lor. - Dezvoltarea abilității de colaborare și comunicare eficientă în echipe multidisciplinare pentru realizarea unui proiect software.

7. Conținutul disciplinei/modulului

Tematica activităților didactice	Numărul de ore
	învățământ cu frecvență
Tematica cursurilor	
1. Introducere. Tipuri de arhitecturi	2
2. Atribute de calitate: disponibilitate, deployability, eficiența energetică	2
3. Atribute de calitate: integrabilitate, modifiabilitate, performanță	4
4. Atribute de calitate: siguranță, securitate, testabilitate, usability	4
5. Arhitectura Monolit	2
6. Arhitectura SOA	4
7. Arhitectura Microservicii	2
8. Arhitectura Event-Driven	4
Total curs:	20
Tematica lucrărilor practice	
1. Introducere în proiectarea arhitecturilor software	4
2. Analiza atributelor de calitate	4
3. Analiza diferențelor dintre arhitecturi	4
4. Realizarea unui proiect simplu folosind arhitectura selectată	8
Total lucrări practice:	20

8. Referințe bibliografice

Principale	1. Richards, Mark. "Fundamentals of Software Architecture." O'Reilly Media, 2020. 2. Bass, Len, et al. "Software Architecture in Practice." Addison-Wesley, 2012.
Suplimentare	1. Rozanski, Nick, și Eoin Woods. <i>Software Systems Architecture: Working with Stakeholders Using Viewpoints and Perspectives</i> . Addison-Wesley, 2012. 2. Garlan, David, et al. "Introduction to Software Architecture." Carnegie Mellon University, 1994.

9. Utilizarea IA generativă

Permișiunea de utilizare	Utilizarea IA generative în cadrul temelor și proiectelor este permisă, cu condiția ca studenții să respecte următoarele reguli: - IA generativă poate fi utilizată pentru generarea de idei, structuri de text sau cod, dar toate materialele generate trebuie să fie revizuite și ajustate de către student pentru a se asigura că acestea corespund cerințelor academice. - Orice utilizare a IA generative trebuie să fie declarată în secțiunea de apendice a fiecărei lucrări, folosind fraza: "În timpul pregătirii acestei lucrări, autorul a utilizat [NUME INSTRUMENT / SERVICIU] în scopul [MOTIV]. După utilizarea acestui instrument/serviciu, autorul a revizuit și editat conținutul după cum a fost necesar și își asumă întreaga responsabilitate pentru conținutul lucrării."
---------------------------------	---

Restricții de utilizare	Studentii nu trebuie să considere IA generativă ca o sursă de încredere pentru informații, deoarece nu oferă referințe clare sau surse documentate. - Nu este permisă citarea directă a conținutului generat de IA în lucrările academice ca și cum ar fi sursă primară. - Activitățile în care este interzis utilizarea IA generativă sunt specificare de profesor și sunt de regulă evaluări intermediare și finale sau care nu presupun activități de dezvoltare a competențelor profesionale.
--------------------------------	---

10. Evaluare

Periodică		Curentă	Studiu individual	Proiect/teză	Examen
EP 1	EP 2				
Învățământ cu frecvență					
15%	15%	15%	15%		40%
Standard minim de performanță					
Prezența și activitatea la prelegeri și lucrări practice.					
Obținerea notei minime de „5” la fiecare dintre atestări și lucrări practice.					

11. Criterii de evaluare

Activitate	Componente evaluare	Metodă de evaluare, criterii de evaluare	Pondere în nota finală a activității	Ponderea în evaluarea disciplinei
Învățământ cu frecvență				
Evaluare periodică I	Conținut teoretic, teme 1-4	Test	100%	15%
Evaluare periodică II	Activitate practică	Dezvoltarea unei aplicații software utilizând o arhitectură la alegere și susținerea orală.	100%	15%
Evaluare curentă	Activitatea practică	Discuții în cadrul seminarelor	50%	15%
		Referat despre o arhitectura software	50%	
Studiul individual	Cercetare la temă	Prezentare/discurs public	100%	15%
Evaluarea finală	Conținut teoretic și practic	Examen oral. Notare conform baremului	100%	40%