

DEZVOLTAREA AVANSATĂ A SISTEMELOR INFORMATIONALE
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	Tehnologia informației				
Anul de studii	Semestrul	Tip de evaluare	Categoria formativă	Categoria de opționalitate	Credite ECTS
Anul I (<i>învățământ cu frecvență</i>)	II	PA	S – uniate de curs de specialitate	O - unitate de curs obligatorie	10

2. Timpul total estimat

Total ore în planul de învățământ		Din care		
		Ore auditoriale	Lucrul individual	
Învățământ cu frecvență	300	Proiect	Studiul materialului teoretic	Pregătire aplicații
		80	80	140

3. Precondiții de acces la disciplină/modul

Conform planului de învățământ	Proiectare conceptuală a unui sistem informational, Arhitectura calculatoarelor, Programarea aplicațiilor distribuite
Conform competențelor	Identificarea și definirea conceptelor și metodelor de procesare a informației necesare dezvoltării aplicațiilor. Înțelegerea etapelor de dezvoltare: analiza cerințelor, proiectare, implementare și testare. Dezvoltarea de soluții software scalabile și robuste. Scrierea de cod modular, reutilizabil și optimizat. Gestionarea bazelor de date și utilizarea eficientă a interfețelor pentru accesarea datelor.

4. Condiții de desfășurare a procesului educațional pentru

Proiect	<ul style="list-style-type: none"> • Pentru prezentarea materialului în sală este nevoie de tablă, tablă interactivă, de proiector și calculator. • Tematica proiectului trebuie să fie aleasă din tematicile disciplinei • Se lucrează individual sau în echipe de 3-4 persoane, fiecare membru își asumă sarcini clare. • Acces la platforme software (IDE-uri, baze de date test) și resurse didactice. • Întâlniri săptămânale pentru raportarea progresului și primirea feedback-ului. • Documentația proiectului să includă specificații tehnice, diagrame de proiectare și raport final. • Demonstrarea soluției și justificarea alegerilor în fața comisiei de evaluare.
----------------	---

5. Competențe specifice acumulate

Competențe profesionale	<p>C3 Privind tehnologiile aplicațiilor</p> <ul style="list-style-type: none"> – Identificarea și definirea conceptelor și metodelor de procesare a informației necesare dezvoltării aplicațiilor. – Capacitatea de a selecta și implementa tehnologii moderne pentru aplicații software. – Adaptarea aplicațiilor la cerințele specifice ale organizațiilor sau utilizatorilor <p>C4 Privind metodele și tehnologiile de dezvoltare software</p> <ul style="list-style-type: none"> – Înțelegerea etapelor de dezvoltare: analiza cerințelor, proiectare, implementare și testare – Dezvoltarea de soluții software scalabile și robuste. – Capacitatea de a analiza probleme complexe și de a defini soluții IT eficiente. – Aplicarea tehnicilor de modelare, inclusiv utilizarea UML sau altor metode vizuale. – Scrierea de cod modular, reutilizabil și optimizat. – Gestionarea bazelor de date și utilizarea eficientă a interfețelor pentru accesarea datelor.
Competențe transversale	<p>CT1. Aplicarea principiilor și normelor de etică în dezvoltarea soluțiilor software. Asigurarea respectării confidențialității datelor și a reglementărilor legale privind utilizarea informațiilor</p> <p>CT2. Dezvoltarea abilității de a lucra eficient în echipă, asumând roluri diferite (lider, membru). Comunicarea clară și colaborarea activă pentru atingerea obiectivelor de proiect. Gestionarea conflictelor și a diferențelor de opinie în mod constructiv.</p> <p>CT3. Capacitatea de a învăța noi tehnologii, metodologii și instrumente. Inițiativa de a ține pasul cu tendințele și schimbările din domeniul tehnologic.</p>

6. Obiectivele disciplinei/modulului

Obiectivul general	Modulul își propune să pregătească studenții în dezvoltarea sistemelor informaționale robuste. Subiectul include dezvoltarea aplicațiilor folosind un sistem de microservicii. Aplicația finală trebuie să fie disponibilă în cloud, aptă pentru scalare orizontală și verticală. Posibilitatea de containerizare și automatizare a proceselor de dezvoltare.
Obiectivele specifice	Se dorește dezvoltarea următoarelor calități: <ul style="list-style-type: none"> • Cunoștințe avansate privind metodologiile structurate de rezolvare a problemelor; • Poate analiza și structura programe scalabile în domeniul de dezvoltare a produselor program; • Propunerea de soluții program scalabile ca membrul unei echipe de dezvoltare. • Poate identifica și aplica platforme de dezvoltare, potrivit obiectivelor proiectului și crea aplicații cu arhitecturi simple, în contextul modulelor corelate (Programarea orientată pe obiecte, Baze de date, Analiza și vizualizarea datelor).

7. Conținutul disciplinei/modulului

Tematica activităților didactice	Numărul de ore
	învățământ cu frecvență
1. Modele și tehnici fundamentale de proiectare	6
2. Înțelegerea naturii problemei, Analiza cerințelor tehnice;	6
3. Paradigme de programare, potrivirea cerințelor	4
4. Arhitectura sistemelor informaționale: monolit, microservicii	6
5. Versionarea sistemului informațional	6
6. Reprezentarea abstractă a proceselor	6
7. Activități esențiale ce fac parte din procesul de dezvoltare avansat	6
8. Metode flexibile (agile) de dezvoltare a soluțiilor program	6
9. Scalabilitatea sistemelor informatice și problemele ce apar în urma lor	6
10. Sisteme de gestiune a bazelor de date	6
11. Interfețe de acces asupra sistemelor de distribuire date	4
12. Tehnici de reutilizare a codului sursă	4
13. Tehnici primare de testare a aplicațiilor	4
14. Cazuri primare de testare	4
15. Tehnici de automatizare și containerizare.	6
Total:	80

8. Referințe bibliografice

Principale	<ol style="list-style-type: none"> 1. D. M. Popovici, I.M. Popovici, J. G. Rican, Proiectarea și implementarea SOFTWARE. București, Teora, 1999. – 238 pag. Biggs J., Tang C. Teaching for Quality Learning at University. 4th Edition, Society for Research into Higher Education & Open University Press, 2011, 391 p. 2. Anca Daniela Ioniță, Modelarea în ingineria sistemelor de programare. București, BIC ALL, 2003. – 207 pag. 3. Dorin Zaharia, Ioan Roșca, Proiectarea obiectuală a sistemelor informaționale. București, DuAl Tech, 2003. – 341 pag. 4. Daniela Saru, Anca D. Ioniță, Sisteme de programe orientate pe obiecte. București, ALL Educational, 2000. – 318 pag. 5. Bruce Eckel, Thinking in Java, Prentice Hall, 2003 8. Niculaescu V., Programarea în Web, București, Jamșa-PRESS, 1998.
Suplimentare	<ol style="list-style-type: none"> 6. D. Bocu, R. Bocu, Modelare obiect orientată cu UML. București, Albastra, 2007 – 251 p 7. David Boud, Nancy Falchikov Aligning assessment with long-term learning. In Journal Assessment & Evaluation in Higher Education Vol. 31, No. 4, August 2006, pp. 399–413 8. H. Yim, et al. Architecture-Centric Object-Oriented Design Method for Multi-Agent Systems. Fourth International Conference on Multi-Agent Systems (ICMAS'00). Los Alamitos, CA, USA:IEEE Computer Society, 2000. ISBN: 0-7695-0625-9.

9. Utilizarea IA generativă

Permisivitatea de utilizare	<p>Utilizarea IA generative în cadrul temelor și proiectelor este permisă, cu condiția ca studenții să respecte următoarele reguli:</p> <ul style="list-style-type: none"> 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 appendice 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	<p>Studenții nu trebuie să considere IA generativă ca o sursă de încredere pentru informații, deoarece nu oferă referințe clare sau surse documentate.</p> <ul style="list-style-type: none"> 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 specificate de profesor și sunt de regulă evaluări intermediare și finale sau care nu presupun activități de dezvoltare a competențelor profesionale.

10. Evaluare

Periodică și curentă		Evaluarea finală
Evaluare 1	Evaluare 2	
30%	30%	40%
Definirea conceptului sistemului informational, Prezentarea unui MVP. Prezentarea diagrame UML pentru definirea functionalitatii sistemului. Containerizarea aplicatiei.		
<p>Prezența și activitatea la seminarele/atelierele de lucru; <i>Obținerea notei „5” la fiecare dintre evaluări;</i> <i>Obținerea notei „5” la lucrarea de examinare finală;</i></p> <p>Evaluarea curentă, fiind de tip formativ și oferind studenților/echipei un feedback continuu la activitățile de proiectare sau modulele integrate, asigură evaluarea studentului cu nota echipei de lucru.</p> <p>Examenul final, fiind o evaluare sumativă, se realizează oral în baza proiectului prezentat public de echipă și discuții/interviuri individuale (în prezența echipei sau nu). Aprecierile obținute la examinare sunt individuale și constituie 40% din nota finală.</p>		

11. Criterii de evaluare

Activitate	Componente evaluare	Metodă de evaluare, criterii de evaluare	Pondere în nota finală a activității	Ponderea în evaluarea disciplinei
Evaluare I	Prezența și participarea la seminar	Observație directă, fișă de prezență	30%	30 %
	Arhitectura sistemului informational	Analiza fiabilitatii arhitecturii selectate intru realizarea proiectului. Definirea granulată a componentelor sistemului	70%	
Evaluare II	Prezentarea sistemului informational 80%.	Analiza functionalizatiilor sistemului informational si relevanta la sarcina propusa. Analiza implementarii arhitecturii.	70%	30 %
	Aplicarea metodologiilor agile intru organizarea dezvoltarii proiectului	Definirea minutioasa a sarcinilor, impartirea pe persoane responsabile, Aproximarea sarcinilor.	30%	
Evaluarea finală	Prezentarea publică a proiectului final	Examinare orală, claritatea expunerii, justificarea deciziilor	70%	40%
	Containerizarea proiectului. Sistemul de versionare. Automatizarea exportarii cloud a proiectului.	Analiza capacitatii proiectului de a se containerize. Automatizarea functionalitatilor de deployment.	30%	