

**D.O.017 ANALIZA ȘI SPECIFICAREA CERINȚELOR SOFTWARE**
**1. Date despre disciplină**

<b>Facultatea</b>	Calculatoare, Informatică și Microelectronică				
<b>Departamentul</b>	Ingineria Software și Automatică				
<b>Ciclul de studii</b>	Studii superioare de licență, ciclul I				
<b>Programul de studii</b>	0613.3 Ingineria Software				
<b>Anul de studii</b>	<b>Semestrul</b>	<b>Tip de evaluare</b>	<b>Categoria formativă</b>	<b>Categoria de opționalitate</b>	<b>Credite ECTS</b>
Anul IV ( <i>învățământ cu frecvență</i> )	7	E	S-Disciplina de specialitate	O - unitate de curs obligatorie	5

**2. Timpul total estimat**

Total ore în planul de învățământ		Din care				
		Ore auditoriale			Lucrul individual	
		Curs	Lucrări practice	Seminar	Studiul materialului teoretic	Pregătire aplicații
<b>Învățământ cu frecvență</b>	<b>150</b>	30	30	15	45	30

**3. Precondiții de acces la disciplină**

<b>Conform planului de învățământ</b>	<p>Pentru a avea acces la această unitate de curs studenții trebuie să fi finalizat cu succes următoarele cursuri, care asigură fundamentul teoretic și practic necesar pentru înțelegerea și aplicarea conceptelor avansate predate:</p> <ul style="list-style-type: none"> <li>Baze de date - acoperă conceptele de bază ale proiectării și gestionării bazelor de date, esențiale pentru gestionarea informațiilor în cadrul aplicațiilor software.</li> <li>Analiza și modelarea sistemelor informaționale - crucial pentru înțelegerea arhitecturii și funcționalității sistemelor informatice complexe și pentru dezvoltarea abilităților de proiectare și modelare a acestor sisteme.</li> <li>Testarea software - asigură cunoștințe esențiale privind tehnicile de testare, identificarea și remedierea erorilor software, contribuind la dezvoltarea de produse software de înaltă calitate.</li> </ul>
<b>Conform competențelor</b>	<p>Accesul la această unitate de curs/modul necesită dobândirea următoarelor competențe, obținute prin finalizarea cursurilor premergătoare menționate mai sus:</p> <ul style="list-style-type: none"> <li>Competențe în gestionarea bazelor de date - necesare pentru proiectarea, implementarea și întreținerea bazelor de date în mediul de lucru profesional.</li> <li>Competențe în analiza și modelarea sistemelor - dobândite prin cursurile de analiză și modelare a sistemelor informaționale, necesare pentru proiectarea arhitecturii sistemelor informatice complexe.</li> <li>Competențe în testarea software – necesare pentru aplicarea tehnicilor de testare, identificarea și remedierea erorilor software, contribuind la asigurarea calității produselor software.</li> </ul>

**4. Condiții de desfășurare a procesului educațional pentru**

<b>Curs</b>	<ul style="list-style-type: none"> <li>Sala de curs, adaptată numărului de studenți în formațiunea de studii (serie), echipată cu proiector și calculator pentru prezentarea materialului teoretic.</li> <li>Nu vor fi tolerate întârzierile studenților, precum și convorbirile telefonice în timpul orelor.</li> </ul>
<b>Lucrări practice/ seminare</b>	<p>Sala destinată <b>desfășurării lucrărilor practice și seminare</b> trebuie să fie dotată cu următoarele echipamente și facilități:</p> <ul style="list-style-type: none"> <li>Prize electrice disponibile pentru ca studenții să poată conecta calculatoarele personale.</li> <li>Conexiune la internet stabilă pentru accesarea resurselor online și desfășurarea activităților interactive.</li> <li>Proiector funcțional pentru prezentarea materialului teoretic și a altor resurse vizuale.</li> </ul> <p><b>Prezentarea lucrărilor practice:</b></p> <ul style="list-style-type: none"> <li>Prezentarea lucrărilor practice este obligatorie și trebuie respectat termenul de predare stabilit. Pentru fiecare lecție întârziată, studentul va fi depunctat cu 1 punct.</li> <li>Lucrarea practică trebuie prezentată sub forma unui raport redactat conform cerințelor specificate pe platforma Moodle și încărcat pe aceeași platformă.</li> <li>Activitatea individuală face parte din procesul de evaluare și trebuie încărcată pe platforma Moodle sub forma unui referat, redactat conform cerințelor stabilite.</li> </ul>

## 5. Competențe specifice acumulate

<b>Competențe profesionale</b>	<p><b>CP1. Elaborarea și proiectarea arhitecturii:</b> studenții vor putea crea și documenta arhitecturi software și hardware pentru soluții IT complexe, respectând cerințele funcționale și nefuncționale.</p> <p><b>CP2. Proiectarea și dezvoltarea aplicațiilor:</b> studenții vor fi capabili să dezvolte aplicații eficiente și scalabile, utilizând metodologii moderne de dezvoltare software și respectând standardele de design.</p> <p><b>CP3. Integrarea componentelor:</b> studenții vor putea integra diferite componente software și hardware într-un sistem unitar, asigurând interoperabilitatea și funcționalitatea optimă a soluției.</p> <p><b>CP4. Testarea aplicațiilor:</b> studenții vor învăța să planifice și să execute teste pentru a valida funcționalitatea, performanța și securitatea aplicațiilor dezvoltate.</p> <p><b>CP5. Implementarea soluțiilor:</b> studenții vor putea implementa soluții IT în medii reale, aplicând cunoștințele dobândite pentru a asigura funcționarea corectă și eficientă a sistemelor.</p> <p><b>CP6. Elaborarea documentației:</b> studenții vor fi capabili să creeze documentația tehnică necesară, inclusiv specificații tehnice, manuale de utilizare și ghiduri de instalare, care să fie clare și ușor de urmărit.</p> <p><b>CP7. Ingineria sistemelor:</b> studenții vor putea proiecta și gestiona sisteme informatice complexe, aplicând principii de inginerie software pentru a îndeplini cerințele proiectului.</p> <p><b>CP8. Managementul problemelor:</b> studenții vor dezvolta abilități de identificare și rezolvare a problemelor apărute în timpul dezvoltării și implementării soluțiilor IT, minimizând impactul acestora asupra proiectului.</p> <p><b>CP9. Îmbunătățirea proceselor:</b> studenții vor putea analiza și optimiza procesele de dezvoltare și implementare a soluțiilor IT, contribuind la îmbunătățirea continuă a performanței și eficienței acestora.</p>
<b>Competențe transversale</b>	<p><b>CT3. Dezvoltare personală și profesională</b>          Conștientizează nevoia de formare continuă cu utilizarea eficientă a resurselor și tehnicilor de învățare pentru dezvoltarea personală și profesională.</p>

## 6. Obiectivele disciplinei

<b>Obiectivul general</b>	<p>De a oferi cunoștințe aprofundate privind managementul cerințelor, cu scopul de a asigura calitatea unui sistem și a unui produs software încă din primele etape ale dezvoltării. Cursul își propune să explice influența cerințelor asupra calității și costului produsului, evidențiind modul în care acestea pot contribui la reducerea costului schimbărilor. Vor fi abordate toate aspectele gestionării cerințelor, precum elicitarea, specificarea, validarea, modelarea și prioritizarea, acordând o atenție specială modelelor de dezvoltare software și diferențelor legate de managementul calității cerințelor. De asemenea, cursul va explora relațiile cu stakeholderii în toate etapele procesului de management al cerințelor.</p>
<b>Obiectivele specifice</b>	<ul style="list-style-type: none"> <li>• Să identifice și să înțeleagă clar elementele care influențează calitatea unui sistem și a unui produs software.</li> <li>• Să cunoască diferitele tipuri de cerințe funcționale și non-funcționale, precum și metodele de identificare a acestora.</li> <li>• Să analizeze punctele forte și punctele slabe ale diferitelor modele de cicluri de viață ale dezvoltării software (SDLC) și efectul acestora asupra procesului de management al cerințelor.</li> <li>• Să înțeleagă întregul proces de management al cerințelor, de la elicitare la specificare, validare și prioritizare.</li> <li>• Să folosească tehnici de analiză pentru cerințele primare, inițiind procesul de elicitare și creare a cerințelor.</li> <li>• Să compare diferitele tehnici de elicitare și să selecteze cea mai potrivită metodă în funcție de contextul și situația dată.</li> <li>• Să înțeleagă importanța stakeholderilor și rolul lor în identificarea și definirea cerințelor.</li> <li>• Să evalueze metodele de negociere cu clienții și stakeholderii privind prioritizarea și extinderea cerințelor.</li> <li>• Să cunoască și să evalueze diferitele tipuri de specificare a cerințelor, identificând avantajele și dezavantajele fiecărei metode pentru a selecta cea mai adecvată abordare.</li> <li>• Să aplice tehnici de asigurare a calității pentru a verifica dacă cerințele sunt verificabile, măsurabile, testabile, exacte, lipsite de ambiguitate, consecvente și complete.</li> <li>• Să gestioneze cererile de modificare a cerințelor și să documenteze aceste modificări în timp.</li> </ul>

## 7. Conținutul disciplinei

Tematica activităților didactice	Numărul de ore
	învățământ cu frecvență
<b>T1. Noțiuni introductive.</b> Definiția defectului. Analiza cauzelor unui defect.	2
<b>T2. Activități de analiză și elaborare a cerințelor</b> Cerințe din perspectiva clientului. Gradul de implicare al participanților în procesul de definire a cerințelor.	2
<b>T3. Niveluri de cerințe</b> Cerințe de domeniu, cerințe de afaceri, cerințe ale clientului și cerințe funcționale. Drepturile și obligațiile clientului. Rolul analistului în elaborarea cerințelor.	4
<b>T4. Standarde pentru elaborarea cerințelor (SRS)</b>	2
<b>T5. Tipuri de cerințe</b> Cerințe cunoscute și necunoscute. Cerințe prescriptive și descriptive. Cerințe nespecificate și/sau nedeclarate.	4
<b>T6. Calitatea cerințelor pentru sistem și software</b> Erori software și impactul acestora. Cauzele erorilor în sistem și software. Indicatori de calitate pentru cerințe de sistem și software. Costul calității cerințelor, ilustrat prin modelul aisbergului.	4
<b>T7. Cerințe non-funcționale</b> Factori de calitate pentru cerințele non-funcționale. Modelele FURPS, Barry W. Boehm și Jim McCall.	2
<b>T8. Ciclul de viață al dezvoltării software (SDLC)</b> Compararea diferitelor modele SDLC. Alegerea modelului SDLC potrivit în funcție de tipul de proiect	2
<b>T9. Metodologii de testare a erorilor fizice și umane pentru un produs TIC</b> Testare funcțională, nefuncțională și structurală. Retestare (retesting), testare de regresie, testare Smoke și Sanity..	4
<b>T10. Gestionarea schimbărilor și asigurarea calității cerințelor</b> Modificarea cerințelor și a produsului software. Planificarea, evaluarea, implementarea și testarea schimbărilor.	2
<b>T11. Documentarea cerințelor pentru sistem și software</b> Șabloane pentru documentarea specificațiilor cerințelor de sistem și software.	2
<b>Total curs:</b>	<b>30</b>
<b>S1. Calitatea începând de la cerințe</b> Impactul cerințelor asupra parcursului proiectului, bugetului și procesului de livrare.	2
<b>S2. Gestionarea stakeholderilor</b> Analiza matricei putere/interes (Power/Interest Grid).Stabilirea stilului de comunicare. Utilizarea CATWOE. Modelul de activitate în afaceri.	2
<b>S3. Cerințe funcționale și non-funcționale</b> Tipuri de cerințe funcționale. Cerințe de calitate.	2
<b>S4. Modelarea cerințelor</b> Evaluarea nivelului adecvat de modelare. Alegerea abordării de modelare pentru diferite tipuri de cerințe.	2
<b>S5. User Stories</b> Crearea de use cases eficiente. Divizarea și structurarea user stories.	2
<b>S6. Ciclul de viață al dezvoltării software (SDLC)</b> Alegerea modelului SDLC potrivit în funcție de cerințele proiectului.	2
<b>S7. Estimare și prioritizare</b> Metode de estimare și prioritizare precum MoSCoW, T-Shirt Sizing, Story Points, NoEstimation, MVP și MAP	2
<b>S8. Controlul modificărilor</b> Practici de control al schimbărilor în funcție de tipul proiectului.	1
<b>Total seminare:</b>	<b>15</b>

Tematica activităților didactice	Numărul de ore
	învățământ cu frecvență
<b>P1. Elicitarea cerințelor</b> Analiza întreprinderii și formularea enunțului problemei pentru un proiect TIC. <ul style="list-style-type: none"> <li>• Analiza obiectului de informatizare</li> <li>• Identificarea și formularea enunțului problemei</li> </ul>	<b>4</b>
<b>P2. Analiza cerințelor</b> Elaborarea unui document SRS (Software Requirement Specification). <ul style="list-style-type: none"> <li>• Colectarea cerințelor</li> <li>• Analiza cerințelor</li> <li>• Crearea documentului de specificații (SRS)</li> </ul>	<b>10</b>
<b>P3. Validarea cerințelor</b> <ul style="list-style-type: none"> <li>• Identificarea, modelarea și validarea cerințelor funcționale pentru proiectul propus</li> <li>• Identificarea și modelarea fluxului de date pentru proiect</li> <li>• Identificarea cerințelor non-funcționale</li> </ul>	<b>10</b>
<b>P4. Managementul cerințelor</b> Planificarea și implementarea cerințelor. <ul style="list-style-type: none"> <li>• Decompoziția activităților pentru implementarea cerințelor</li> <li>• Planificarea realizării cerințelor</li> <li>• Estimarea costurilor pentru implementarea cerințelor</li> </ul>	<b>6</b>
<b>Total practice:</b>	<b>30</b>

## 8. Referințe bibliografice

<b>Principale</b>	<ol style="list-style-type: none"> <li>1. P. Chirev, „FCIM.ASCS21.1”. Disponibil la: <a href="https://else.fcim.utm.md/enrol/index.php?id=794">https://else.fcim.utm.md/enrol/index.php?id=794</a>.</li> <li>2. D. P. Michelfelder și N. Doorn, Ed., The Routledge Handbook of the Philosophy of Engineering, 1 ed. Routledge, 2020. doi: 10.4324/9781315276502. Disponibil la: <a href="https://www.taylorfrancis.com/books/9781351996563">https://www.taylorfrancis.com/books/9781351996563</a>.</li> <li>3. M. Cohn, User stories applied: for agile software development. în Addison-Wesley signature series. Boston: Addison-Wesley, 2004.</li> <li>4. K. Pohl și C. Rupp, Requirements engineering fundamentals: a study guide for the certified professional for requirements engineering exam, foundation level, IREB compliant, Second edition. Santa Barbara, CA: Rocky Nook, 2015.</li> <li>5. S. Robertson și J. Robertson, Mastering the requirements process: getting requirements right, Third edition. Upper Saddle River, NJ: Addison-Wesley, 2012.</li> <li>6. JK. E. Wiegers și J. Beatty, Software requirements, Third edition. Redmond, Washington: Microsoft Press, s division of Microsoft Corporation, 2013.</li> </ol>
<b>Suplimentare</b>	<ol style="list-style-type: none"> <li>1. J. Ludewig și H. Lichter, Software Engineering: Grundlagen, Menschen, Prozesse, Techniken, 4., Überarbeitete und erweiterte Auflage. Heidelberg: dpunkt.verlag, 2023.</li> <li>2. A. Trendowicz și R. Jeffery, Software Project Effort Estimation: Foundations and Best Practice Guidelines for Success. Cham: Springer International Publishing, 2014. doi: 10.1007/978-3-319-03629-8. Disponibil la: <a href="https://link.springer.com/10.1007/978-3-319-03629-8">https://link.springer.com/10.1007/978-3-319-03629-8</a></li> </ol>

## 9. Utilizarea IA generativă

<b>Permișunea de utilizare</b>	Utilizarea IA generative în cadrul temelor și proiectelor este permisă, cu condiția ca studenții să respecte următoarele reguli: <ul style="list-style-type: none"> <li>• 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.</li> <li>• 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."</li> </ul>
--------------------------------	---

<b>Restricții de utilizare</b>	<p>Studentii 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"> <li>Nu este permisă citarea directă a conținutului generat de IA în lucrările academice ca și cum ar fi sursă primară.</li> <li>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.</li> </ul>

### 10. Evaluare

Periodică		Curentă	Lucrul individual	Examen
EP 1	EP 2			
<b>Învățământ cu frecvență</b>				
15%	15%	15%	15%	40%
Standard minim de performanță. Prezența și activitatea la prelegeri și lucrări practice. Prezentarea lucrului individual. Obținerea notei minime de „5” la fiecare lucrări practice și lucrul individual.				

### 11. Criterii de evaluare

Activitate	Componente evaluare	Metodă de evaluare, Criterii de evaluare	Pondere în nota finală a activității	Ponderea în evaluarea disciplinei
<b>Învățământ cu frecvență</b>				
<b>Evaluare periodică I</b>	Conținut teoretic, teme 1-5	Test pe MOODLE	100%	<b>15%</b>
<b>Evaluare periodică II</b>	Conținut teoretic, teme 6-10	Test pe MOODLE	100%	<b>15%</b>
<b>Evaluare curentă</b>	Activitatea practică	Discuții în cadrul orelor de practică	50%	<b>15%</b>
		Raport pentru fiecare lucrare de practică încărcat pe MOODLE	50%	
<b>Lucrul individual</b>	Cercetare la temă	Referat/Prezentare/discurs public. Raportul încărcat pe MOODLE	100%	<b>15%</b>
<b>Evaluarea finală</b>	Conținut teoretic și practic	Test pe MOODLE	100%	<b>40%</b>