

FIȘA DISCIPLINEI

1. Date despre program

1.1. Instituța de învățământ superior	Universitatea „Tibiscus” din Timișoara
1.2. Facultatea	Calculatoare și Informatică Aplicată
1.3. Departamentul	Informatică
1.4. Domeniul de studii	Informatică
1.5. Ciclul de studii	Licență
1.6. Programul de studii/Calificarea	Informatică / Asistent de cercetare în informatică - 214918; Proiectant sisteme informatice - 25110; Analist - 251201; Administrator baze de date - 252101; Administrator de rețea de calculatoare - 252301; Profesor în învățământul gimnazial - 233002; Programator - 251202

2. Date despre disciplină

2.1. Denumirea/codul disciplinei	INGINERIE SOFTWARE (IS) - (OPT.3) LIN233						
2.2. Titularul activității de curs	Lect.univ.drd. Valentin Lucian CIORBA						
2.3. Titularul activității de laborator	Lect.univ.drd. Valentin Lucian CIORBA						
2.4. Anul de studiu	2	2.5. Semestrul	1	2.6. Tipul de evaluare	E	2.7. Regimul disciplinei	DS

3. Timpul total estimat

3.1. Numărul de ore pe săptămână	3	din care 3.2. curs	1	3.3. laborator	2
3.4. Total ore din planul de învățământ	42	din care 3.5. curs	14	3.6. laborator	28
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					10
Documentare suplimentară în bibliotecă, pe platforme electronice de specialitate și pe teren					24
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					24
Tutoriat					4
Examinări					4
Alte activități					
3.7. Total ore studiu individual					58
3.8. Total ore pe semestru					100
3.9. Numărul de credite					4

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Arhitectura calculatoarelor
4.2. de competențe	Programare, Modelare UML, Cunoștințe fundamentale de inginerie software

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Online: Google Classroom, Zoom
5.2. de desfășurare a laboratorului	Online: Google Classroom, Zoom

6. Competențe specifice acumulate

6.1. Competențe profesionale	C1. Explicarea unor aplicații soft existente, pe niveluri de abstractizare (arhitectură, pachete, clase, metode) utilizând în mod adecvat cunoștințele de bază,
------------------------------	---

	C2. Capacitatea de a dezvolta aplicații software, C3. Identificarea și explicarea mecanismelor adecvate de specificare a sistemelor software.
6.2. Competențe transversale	CT1. Desfășurarea eficientă a activităților organizate într-un grup inter-disciplinar și dezvoltarea capacităților empatică de comunicare inter-personală, de relaționare și colaborare cu grupuri diverse, CT2. Capacitatea de a se familiariza cu noi concepte și de a se adapta rapid la noile tehnologii ce apar în domeniul informaticii.

7. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

7.1. Obiectivul general al disciplinei	<ul style="list-style-type: none"> ● Dezvoltarea de aplicații software, ● Cunoașterea metodelor de lucru în echipă și a etapelor de dezvoltarea a aplicațiilor software, ● Dezvoltarea de abilități de rezolvare a problemelor, de a analiza o problemă reală venită din partea potențialilor clienți și de a dezvolta soluții prin proiectare software, folosind tehnologii moderne.
7.2. Obiectivele specifice	<ul style="list-style-type: none"> ● Însușirea principalelor concepte care operează în cadrul activității de proiectare software. ● Însușirea cunoștințelor și deprinderilor practice necesare lucrului într-o echipă de proiectare software. ● Dobândirea unor cunoștințe pentru abordarea cu succes a proiectelor software complexe.

8. Conținuturi

8.1. Curs	Metode de predare	Observații
1. Introducere în Ingineria programării	Expunerea interactivă, problematizarea, conversația euristică, documentarea pe web, exemplificarea	1 oră
2. Limbajul unificat de modelare, UML		1 oră
3. Faza de analiză		1 oră
4. Faza de proiectare		1 oră
5. Șabloane de proiectare creaționale (I)		2 ore
6. Șabloane de proiectare structurale (I)		2 ore
7. Șabloane de proiectare comportamentale (I)		2 ore
8. Faza de implementare		1 oră
9. Faza de testare (I)		2 ore
10. Aspecte conexe procesului de dezvoltare		1 oră

Bibliografie

- [1]. Florin Leon, *Ingineria programării*, http://florinleon.byethost24.com/curs_ip.htm?i=1
[2]. Ian Sommerville, *Software Engineering*, 8th Edition, Addison Wesley, 2006
[3]. K. Schwalbe, *Information Technology Project Management*, Fifth Edition, 2007, Course Technology, CENGAGE Learning
[4]. Ian Sommerville, "Software Engineering" Eighth Edition, Addison-Wesley, 2007.
[5]. Scott W. Ambler, "The Elements of UML 2.0 Style", Cambridge University Press, 2005.

8.2. Laborator	Metode de predare	Observații
1. Arduino web editor	Exercițiul, discuțiile și dezbaterile, modelarea, proiectarea și realizarea unei aplicații, lucrul în grup organizat	3 săptămâni – 6 ore
2. Visual Paradigm		3 săptămâni – 6 ore
3. Diagrame de secvențe UML cu draw.io		3 săptămâni – 6 ore
4. Diagrame de relații a entităților cu draw.io		3 săptămâni – 6 ore
5. draw.io Privacy și Google Analytics		2 săptămâni – 4 ore

Bibliografie

- <https://www.visual-paradigm.com/>
- <https://about.draw.io/uml-class-diagrams-in-draw-io/>
- <https://about.draw.io/create-uml-sequence-diagrams-in-draw-io/>
- <https://about.draw.io/draw-io-privacy-and-google-analytics/>

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

Cunoașterea problematicii și a managementului proceselor în cadrul activităților de dezvoltare software este necesară oricărui candidat la angajare într-o firmă de dezvoltare de software. De asemenea, firmele client pentru produse de software pot beneficia de pregătirea unui inginer software.

10. Evaluare

Tip de activitate	10.1. Criterii de evaluare	10.2. Metode de evaluare	10.3. Pondere din nota finală
10.4. Curs	- Cunoașterea problematicii prezentate și discutate la curs - Cunoștințe generale, evaluate printr-un test cuprinzând întrebări cu variante multiple de răspuns.	Examinare scrisă online	30%
10.5. Laborator	Realizarea unui proiect pe o temă primită / aleasă	Evaluarea activității și participarea activă la laborator Prezentarea proiectului realizat	70%
10.6. Standard minim de performanță			
Cunoașterea noțiunilor, aspectelor fundamentale teoretice. Abilități de utilizare a proceselor și a tehnologiilor specifice dezvoltării software. Utilizarea unor instrumente software pentru managementul proceselor în ingineria software.			


Data completării

28.09.2020

Semnătura titularului de curs



Semnătura titularului de laborator



Data avizării în departament

05.10.2020

Semnătura directorului de departament

