

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Transilvania din Brașov	
1.2 Facultatea	Matematică și Informatică	
1.3 Departamentul	Matematică și Informatică	
1.4 Domeniul de studii de ¹⁾	Informatică	
1.5 Ciclul de studii ²⁾	Masterat	
1.6 Programul de studii/ Calificarea	Aplicatii pentru telefonul mobil si tehnologii Internet in e-busir	

2. Date despre disciplină

2.1 Denumirea disciplinei	ARHITECTURA SOFTWARE							
2.2 Titularul activităților de curs	Dr.-Ing.Detlef STREITFERDT							
2.3 Titularul activităților de seminar/ laborator/ proiect	Dr.-Ing.Detlef STREITFERDT							
2.4 Anul de studiu	I	2.5 Semestrul	1	2.6 Tipul de evaluare	C	2.7 Regimul disciplinei	Conținut ³⁾	DAP
							Obligativitate ⁴⁾	DO

3. Timpul total estimat (ore pe semestru al activităților didactice)

3.1 Număr de ore pe săptămână	3	din care: 3.2 curs	2	3.3 seminar/ laborator/ proiect	0/1/0
3.4 Total ore din planul de învățământ	42	din care: 3.5 curs	28	3.6 seminar/ laborator/ proiect	0/14/0
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					45
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					25
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					30
Tutoriat					4
Examinări					4
Alte activități.....					-
3.7 Total ore de activitate a studentului	108				
3.8 Total ore pe semestru	150				
3.9 Numărul de credite ⁵⁾	6				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Software Engineering la nivel de Bachelor
4.2 de competențe	- Cunoașterea proceselor de dezvoltare software - Modelare orientată pe obiecte - Programare orientată pe obiecte

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	Calculatoare pe care sa fie instalate mai multe browsere (Chrome, Firefox).
5.2 de desfășurare a seminarului/ laboratorului/ proiectului	Calculatoare pe care sa fie instalate mai multe browsere (Chrome, Firefox).

6. Competențe specifice acumulate (conform grilei de competențe din planul de învățământ)

Competențe profesionale	C1. Gestionează conținut online, transpune cerințele într-un model vizual, utilizează interfețe pecifice aplicațiilor R.Î.1.1 Absolventul se asigură ca conținutul site-ului web este la zi, organizat, atractiv și satisface nevoile publicului-tintă, cerințele societății și standardele internaționale, prin verificarea linkurilor, prin stabilirea intervalului de publicare și a ordinii. R.Î.1.2 Absolventul elaborează modelul vizual pornind de la specificații și cerințe date, pe baza analizei domeniului de aplicare și a publicului-tintă. R.Î.1.3 Absolventul creează o reprezentare vizuală a ideilor, cum ar fi sigle, elemente grafice pentru site-ul web, jocuri și formate digitale. C2. Creează scheletul website-ului, interacționează cu utilizatorii pentru a le afla cerințele, stabilește relații de afaceri, definește cerințe tehnice R.Î.2.1 Absolventul elaborează o imagine sau un set de imagini care afișează elementele funcționale ale unui site web sau ale unei pagini, utilizate în mod obișnuit pentru planificarea funcționalității și structurii unui site internet. R.Î.2.2 . Absolventul stabilește o relație pozitivă, pe termen lung, între organizații și părțile terțe interesate, cum ar fi furnizorii, distribuitorii, acționarii și alte părți interesate, pentru a-i informa cu privire la organizație și obiectivele ei. R.Î.2.3. Absolventul specifică proprietățile tehnice ale marfurilor, materialelor, metodelor, proceselor, serviciilor, sistemelor, software-ului și funcționalităților, prin identificarea și abordarea nevoilor specifice care trebuie satisfăcute conform cerințelor clientului.
Competențe transversale	CT1 . Aplica cunoștințe științifice, tehnologice și ingineresti; respecta diversitatea valorilor și a normelor culturale. R.Î.1.1 Absolventul adoptă tehnologii noi. R.Î.1.2 Absolventul ține cont de diversitatea valorilor și normelor culturale CT2 Aplica măsuri de securitate digitală; are o minte deschisă. R.Î.1.1 Absolventul protejează dispozitivele TIC și elimină virusii și programele malware de pe un computer R.Î.1.2 Absolventul implementează măsuri de securitate digitală R.Î.1.3 Absolventul preia idei și experiențe noi

7. Obiectivele disciplinei (reieșind din competențele specifice acumulate)

7.1 Obiectivul general al disciplinei	• Munca de cooperare • Metode de dezvoltare a unui proiect
7.2 Obiectivele specifice	• Stiluri de arhitectură • Modele de design

8. Conținuturi

8.1 Curs	Metode de predare	Număr de ore	Observații
1. Introducere - Definiția „Arhitectura software” - Alte cerințe pentru arhitecturile software - De la cerințe la design	Curs interactiv, expunere, proiector. Învățare prin probleme/proiecte Conversația dezbaterii, dialogul pentru fixarea și a consolidarea a noțiunilor prezentate	4	
2. Procese de dezvoltare software - Modelul cascadă, prototip, evolutiv, spirală - OpenUP - SCRUM		4	
3. Începerea proiectului - WWW - Stakeholderi, interviuri, brainstorming		4	

- Cercetare de brevete - Instrumente RE			
4. Exemple de standarde în dezvoltarea software - IEC61508 - EN50128		4	
5. Stiluri de arhitectură cu exemple - Straturi - Pipes & Filters - Blackboard - Broker - Model-View-Controller - Microkernel - Reflection - Exemplu: CycleComputer - Exemplu: Android - Exemplu: Pong - Exemplu: Bourne Again Shell - EN50128, Arhitectură - Modele de design - Șablon de documentare ARC42		4	
6. Analiza arhitecturii. Metrice de cod		4	
7. Testare . Testare EN50128		4	
Bibliografie - Robert C. Martin,,Clean Architecture - A Craftsman's Guide to Software Structure and Design",Prentice Hall,201B, - Joseph Ingeno,,Software Architect's Handbook", Packt Publish in g, 20 1 B Neal Ford, Rebecca Parsons, Patrick Kua, „Building Evolutionary Architectures: Support Constant Change", O-Reilly Safari, 2017. - Rick Kazman, Humberto Cervantes, „Designing Software Architectures: A Practical Approach", Addison-Wesley Professional, 2016. - Len Bass,,Software Architecture inPractice 3rd Edition", Pearson Education,2013. Stephen Withall, "Software Requirement Patterns", Microsoft Press, 2007 [Balz 1996] Helmut Balzert, "Lehrbuch der Software-Technik", Spektrum Akademischer Verlag, 1996, [Fowl 1999] Martin Fowler, "Refactorng - Improving the Design of Existing Code", Addison Wesley, 1999, [Gamm 1995] Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design Patterns - Elements of Reusable Object-Oriented Software", Addison Wesley, 1 995. [Mart 2009] Robert C. Martin, „Clean Code", Prentice Hall, 2009. [Rooc 2004] Steve McConnell, „Code Complete 2nd Edition", Microsoft Press, 2004 [Rooc 2004] Stefan Roock, Martin Lippert, "Refactorings in groBen Softwareprojekten", dpunkt.Verlag GmbH,2004 [Somm 2007] Ian Sommerville,"software Engineering", Pearson Studium, 2007, [Mens 2008] T. Mens and S. Demeyer, Eds., Software Evolution. Springer-Verlag New York 1nc,2008 [Bere 2009] Brian Berenbach, Daniel J. Paulish, Juergen Kazmeier, Arnold Rudorfer, "Software & Systems Requirements Engineering In Practice", Mc Graw Hill, 2009. [Lams 2001]Axel van Lamsweerde, "Goal-Oriented Requirements Engineering: A Guided Tour", in Proceedings of the 5th IEEE International Symposium on Requirements Engineering (RE 2001),27-31 August 2001, Toronto, Canada, 2001. [Lams 2009] Axel van Lamsweerde, "Requirements Engineering: From System Goals to UML IVodelsto Software Specifications", John Wiley & Sons, 2009. [McCo 2006] Steve McConnell, "software Estlmination", Microsoft Press, 2006. [Pohl 2008] Klaus Pohl, "Requirements Engineering: Grundlagen, Prinzipien,Techniken", dpunkt,Verlag GmbH,2008. [Boec 2004] Gunter Bockle, Peter Knauber, Klaus Pohl, Klaus Schmid, "Software-Produktlinien: Methoden, Einfuhrung und Praxis", dpunkt.Verlag GmbH, 2004 [- Clem 2002] Paul Clements, Rick Kazman, Mark Klein, "Evaluating Software Architectures", Addison Wesley,2002. [Hrus 201 2] P. Hruschka and G, Starke, Architektur-Knigge fur Softwarearchitekten-Der Verschatzer, 201 2,			
8.2 Seminar/ laborator/ proiect	Metode de predare-învățare	Număr de ore	Observații
1. Adaptarea unui proces de dezvoltare software		2	

2. Exercițiu: cu propria cercetare de brevete	Învățarea prin probleme / proiecte.	5	
3. Colectarea cerințelor - exercitii		2	
4. Dezvoltarea unei arhitecturi software proprii		3	
5. Proiect propriu		2	
Bibliografie Vezi mai sus			

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunităților epistemice, ale asociațiilor profesionale și ale angajatorilor reprezentativi din domeniul aferent programului

Conținuturile sunt în concordanță cu problematica monografiilor apărute în ultimii ani în literatura de specialitate

10. Evaluare

Tip de activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Însușirea și aplicarea noțiunilor predate la curs	Rezolvarea de exerciții și probleme	10%
	Examen grila		20%
10.5 Seminar/ laborator/ proiect	Însușirea lucrărilor de laborator	Realizarea unor aplicații asemănătoare celor facute în cadrul orelor de laborator	20%
	Proiect	Verificarea realizării proiectului	50%
10.6 Standard minim de performanță			
Obținerea notei 6 la toate tipurile de evaluare			

Prezenta Fișă de disciplină a fost avizată în ședința de Consiliu de departament din data de 26/09/2024 și aprobată în ședința de Consiliu al facultății din data de 26/09/2024

Conf.dr. Ion-Gabriel STAN, Decan	Conf.dr. Nicusor MINCULETE Director de departament
Dr.-ing.Detlef STREITFERDT Titular de curs	Dr.-ing.Detlef STREITFERDT Titular de seminar/ laborator/ proiect

Notă:

¹⁾ Domeniul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat (se completează conform cu Nomenclatorul domeniilor și al specializărilor/ programelor de studii universitare în vigoare);

²⁾ Ciclul de studii - se alege una din variantele: Licență/ Masterat/ Doctorat;

- ³⁾ Regimul disciplinei (conținut) - se alege una din variantele: DF (disciplină fundamentală)/ DD (disciplină din domeniu)/ DS (disciplină de specialitate)/ DC (disciplină complementară) - pentru nivelul de licență; DAP (disciplină de aprofundare)/ DSI (disciplină de sinteză)/ DCA (disciplină de cunoaștere avansată) - pentru nivelul de masterat;
- ⁴⁾ Regimul disciplinei (obligativitate) - se alege una din variantele: DI (disciplină obligatorie)/ DO (disciplină opțională)/ DFac (disciplină facultativă);
- ⁵⁾ Un credit este echivalent cu 25 de ore de studiu (activități didactice și studiu individual).