

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Facultatea de Electronică, Telecomunicații și Tehnologia Informației
1.3 Departamentul	Comunicații
1.4 Domeniul de studii	Inginerie electronică, telecomunicații și tehnologii informaționale
1.5 Ciclul de studii	Master
1.6 Programul de studii / Calificarea	Telecomunicații (TC) / Master
1.7 Forma de învățământ	IF – învățământ cu frecvență
1.8 Codul disciplinei	TC16.40

2. Date despre disciplină

2.1 Denumirea disciplinei	Elemente software avansate in telecomunicatii						
2.2 Aria de conținut	Arie teoretică						
	Arie metodologică						
	Arie de analiză						
2.3 Responsabil de curs	Prof.dr.ing. Mircea-Florin VAIDA – Mircea.Vaida@com.utcluj.ro						
2.4 Titularul activităților de seminar / laborator / proiect	Prof.dr.ing. Mircea-Florin VAIDA – Mircea.Vaida@com.utcluj.ro						
2.5 Anul de studiu	2	2.6 Semestrul	3	2.7 Tipul de evaluare	E	2.8 Regimul disciplinei	DS/DO

3. Timpul total estimat

3.1 Număr de ore pe săptămână	3	din care: 3.2 curs	2	3.3 seminar / laborator	1
3.4 Total ore din planul de învățământ	42	din care: 3.5 curs	28	3.6 seminar / laborator	14
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					10
Pregătire seminarii / laboratoare, teme, referate, portofolii și eseuri					10
Tutoriat					10
Examinări					4
Alte activități: proiect cercetare					4
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	Programarea calculatoarelor – Limbaje, Algoritmi. Ingineria Programarii.
4.2 de competențe	Concepte de baza privind dezvoltarea aplicatiilor software, concepte de programare obiectuala, algoritmi si tehnici de programare, elemente de baza privind ingineria programarii. Abilitati de a utiliza un mediu integrat (Visual Studio:C++/C#, Eclipse-Java, etc.)

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Video-proiector, ecran, white board, tabla neagra
5.2. de desfășurare a seminarului / laboratorului / proiectului	PC-uri cu acces Internet

6. Competențele specifice acumulate

Competențe profesionale	C3 Aplicarea cunoștințelor, conceptelor și metodelor de bază privitoare la arhitectura sistemelor de calcul, microprocesoare, microcontrolere, limbaje și tehnici de programare C4. Conceperea, implementarea și operarea serviciilor de date, voce, video, multimedia, bazate pe înțelegerea și aplicarea noțiunilor fundamentale din domeniul comunicațiilor și transmisiunii informației C5. Selectarea, instalarea, configurarea și exploatarea echipamentelor de telecomunicații fixe sau mobile și echiparea unui amplasament cu rețele uzuale de telecomunicații
Competențe transversale	N/A

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

7.1 Obiectivul general al disciplinei	Dezvoltarea de competențe profesionale în domeniul aplicațiilor software
7.2 Obiectivele specifice	Modele, metodologii software. Dezvoltare de software generic și avansat. Software testing. Elemente de baza privind dezvoltarea de software în cloud.

8. Conținuturi

8.1 Curs	Metode de predare	Observații
1. Modele software de creare a aplicațiilor în telecomunicații. Ciclul de viață al programelor și al sistemelor.	Expunere la tablă, prezentare cu videoproiector, discuții.	Nu este cazul.
2. Metodologii de clasificare, structurale: SSADM-MERISE, obiectuale:OOD,OOT și formale. Interfatarea și implementarea. Principiul lui Parnas. Metodologii moderne de gestiune a aplicațiilor software: Agile/Scrum și Just in time/Kanban		
3. Factorii umani și implicațiile lor în programare. Sisteme informaționale, IS și tehnologia informației, IT. Psihologia și memoria umană în procesul IP. Nevoile utilizatorilor. Metodologii de proiectare IS.		
4. Metodologii educaționale alternative în vederea dezvoltării aplicațiilor software (creare echipe de lucru compatibile).		
5. UML. Diagrame de colaborare- conexiuni- legături -interacțiuni: repetitive, condiționale, multithreading, precondiții, sincrone, asincrone. Diagrame de stare. Diagrame de punere în funcțiune. Ciclul de viață al componentelor software în Metodologia OO (OOM). Studiu de caz.		
6. Noi facilități oferite de biblioteca C++0x/1y/2z. Noi elemente introduse în limbajul C++0x/1y/2z.		
7. Concepte privind evoluția programării generice: funcții/metode și		

clase template in C++. Evolutie si utilizare.		
8. Biblioteca STL. Definire si utilizare.		
9. Programarea generica in Java. Colectii, evolutie si utilizare.		
10. Programarea multithreading si sincronizarea in Java.		
11. Programarea paralela/multicore. Concepte, utilizare in Java.		
12. Interfete grafice in programare. Design Pattern-uri. Exemplificare MVC.		
13. Cloud si Swarm computing. Introducere, comparație tehnologii existente.		
14. Testarea manuala si automata a aplicatiilor software. JUnit si alte facilitati de testare a aplicatiilor software.		
Bibliografie		
1. Mircea-Florin Vaida, Cosmin Porumb, Radu Fotea, Florin Hurducas, Liviu Lazar, Java 2 Enterprise Edition (J2EE). Aplicatii multimedia, Editura Albastra, 2003		
2. Cosmin Strilețchi, Mircea-F. Vaida, Ligia-D. Chiorean, Adriana Stan, Noțiuni esențiale și tehnologii specifice limbajului Java, Casa Cartii de Stiinta, 2018		
3. B. Stroustrup, The C++ programming language, Addison-Wesley, 2015		
4. S. Tanasa, C. Olariu, Dezvoltarea aplicatiilor Web folosind Java, Ed. Polirom 2005		
5. L. Alboaie, S. Buraga, Servicii Web. Concepte de baza si implementari, Ed. Polirom 2006		
6. Mircea-Florin Vaida, Ligia-Domnica Chiorean, Lenuța Alboaie, Petre Gavril Pop, Cosmin Strilețchi, Kuderna-Iulian Bența, Programarea în limbajul C/C++ cu elemente C++1y. Programare web C++, Casa Cartii de Stiinta, Cluj-Napoca, 2016		
7. Ligia-Domnica Chiorean, Kuderna-Iulian Bența, Mircea-Florin Vaida, Petre Gavril Pop, Cosmin Strilețchi, C/C++ - Ghid teoretic si practic, Casa Cartii de Stiinta, Cluj-Napoca, 2016		
8. Cosmin Strilețchi, Ligia-D. Chiorean, Mircea-F. Vaida, Adriana Stan, Stefan Dragos, Tehnologii Java orientate spre aplicații cross-platform, Casa Cartii de Stiinta, 2020		
9. Materiale suport de curs/laborator, evaluare, indrumare pe platforma Helios a colectivului de software, https://helios.utcluj.ro/learn2code/index.php		
8.2 Laborator	Metode de predare	Observații
1. Principiile si obiectivele Inginerii Programarii reflectate in dezvoltarea aplicatiilor software		
2. Elemente fundamentale de programare a aplicatiilor folosind un limbaj obiectual (C/C++, C#, Java). Elemente specifice avansate in dezvoltarea aplicatiilor obiectuale.		
3. Metodologii educationale alternative. Grupare in echipe de lucru folosind tipologii ale enegramei si MBTI.		
4. Metodologii de gestiune/ proiectare a aplicatiilor software tinand cont de specificatii standard de firma.- Redactarea si evaluarea unui raport stiintific implicand dezvoltarea aplicatiilor software avansate. Definire teme pentru echipe. Redactare articole in reviste si conferinte. Utilizarea diagramelor UML in procesul software. - Definire mecanisme creare echipe de lucru.		
5. Dezvoltarea de aplicatii software folosind C++0x/1y/2z si clase si functii/metode template in C++ (generice C++). Dezvoltarea de aplicatii software folosind STL. Etapa intermediara pre-evaluari echipe de lucru- stabilirea caiet sarcini final.		
6. Dezvoltarea de aplicatii software folosind folosind Java Generics. Multithreading in Java. Fork-join in Java pentru		

programarea paralela/multicore. Testare aplicatii folosind JUnit. – Evaluari echipe de lucru- state of art, tehnologii utilizate		
7. Evaluare teme laborator. Prezentarea unui raport stiintific pe o tematica software conform unor specificatii standard de firma. Evaluari activitate echipe.		
Bibliografie Comuna cu cursul On-line Platforms: <ul style="list-style-type: none"> Zoom: MirceaF Vaida: Personal Meeting ID 2412138787 https://zoom.us/j/2412138787 pwd: https://us04web.zoom.us/j/2412138787?pwd=dnNSaGJKY0VkYlJVV1BuL1ZsYXpaUT09 Microsoft Teams: Class: ESATC-ITM, Code: wmbt10y 		

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

<p>Competențele dobândite vor fi folosite în următoarele ocupații conform COR (Clasificarea Ocupațiilor din România): Inginer emisie; Inginer electronist, transporturi, telecomunicații; Inginer imagine; Inginer sunet; Proiectant inginer electronist; Proiectant inginer de sisteme și calculatoare; Inginer șef car reportaj; Inginer șef schimb emisie; Inginer proiectant comunicații; Inginer sisteme de securitate; Inginer suport vânzări; Dezvoltator de aplicații multimedia; Inginer operare rețea; Inginer testare sisteme de comunicații; Manager proiect; Inginer de trafic; Consultant pentru sisteme de comunicații.</p>
--

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Test scris cu întrebări. Elaborare raport/lucrare științifică	Examen scris/oral (T=25%) + Activitate științifică în timpul semestrului (S=25%) E = T + S	E=50%
10.5 Seminar/Laborator	Activitate practică în echipă dezvoltată la laborator și ACP	Activitate practică (P)	P=50%
10.6 Standard minim de performanță			
Nivel calitativ: <i>Cunoștințe minimale:</i> <ul style="list-style-type: none"> ✓ Răspuns corect la cel puțin un subiect de teorie, scrierea relațiilor de proiectare adecvate necesare pentru rezolvarea problemei și obținerea unei note minime de 5 în cadrul activităților de laborator. <i>Competențe minimale: Studentii să știe,</i> <ul style="list-style-type: none"> ✓ Să facă distincție între dezvoltarea aplicațiilor software didactice și profesionale; ✓ Să utilizeze metodologii moderne de dezvoltare a aplicațiilor software în echipe definite pe compatibilități de tip Agile/Kanban; ✓ Să utilizeze metodologii de proiectare software obiectuale; ✓ Să utilizeze programarea generică, multithreading, paralela/multicore ✓ Să utilizeze metodologii de proiectare software bazate pe diagrame UML; ✓ Să înțeleagă și să cunoască rolul testării aplicațiilor software; ✓ Să dezvolte aplicații folosind noile facilități C++0x/1y/2z; ✓ Să cunoască elemente privind dezvoltare de software în cloud. Nivel cantitativ: <ul style="list-style-type: none"> ✓ Efectuarea tuturor lucrărilor de laborator ✓ Notele la examen și laborator să fie minim 5. 			

✓ Nota la disciplină se calculează cu relația: $0,5 * \text{Nota_examen} + 0,5 * \text{Nota_laborator}$

Data completării:	Titulari	Titlu Prenume NUME	Semnătura
19.06.2024	Curs	Prof.dr.ing. Mircea-Florin VAIDA	
	Aplicații	Prof.dr.ing. Mircea-Florin VAIDA	

Data avizării în Consiliul Departamentului COM 10.07.2024	Director Departament Comunicații Prof.dr.ing. Virgil DOBROTĂ
Data aprobării în Consiliul Facultății ETTI 11.07.2024	Decan Prof.dr.ing. Ovidiu POP