FIŞA DISCIPLINEI

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea „Sapientia" din municipiul Cluj-Napoca
1.2. Facultatea/ DSPP	Facultatea de Ştiințe Tehnice şi Umaniste din Târgu Mureş
1.3. Domeniul de studii	Calculatoare şi tehnologia informației
1.4. Ciclul de studii	Licență
1.5. Programul de studiu	Calculatoare
1.6. Calificarea	Inginer

2. Date despre disciplină

2.0. Departamentul				Inginerie electrică			
2.1. Denumirea disciplinei				Sisteme distribuite			
2.2. Titularul disciplinei / a activităților de curs				Dr. Vajda Tamás			
2.3. Titularul (ii) activităților de	seminar						
	laborator			Dr. Vajda Tamás			
	proiect			Dr. Vajda Tamás			
2.4. Anul de studiu	3	2.5. Semestrul	6	2.6. Tipul de evaluare	Ex	2.7. Regimul disciplinei	DI

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

3.1. Număr de ore pe săptămână	4	Din care: 3.2. curs	2	3.3. seminar / laborator/ proiect/ practica	0/ 1/ 1/ 0
3.4. Total ore din planul de învățământ	28	Din care: 3.5. curs	14	3.6. seminar/ laborator/ proiect/ practica	0/ 14/ 14/ 0

Distribuția fondului de timp:	ore
a) Studiul după manual, suport de curs, bibliografie şi notițe	**10**
b) Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren	10
c) Pregătire laboratoare, teme, referate, portofolii şi eseuri	22
d) Tutoriat	
e) Examinări	2
f) Alte activități:	

3.7. Total ore studiu individual	44
3.8. Total ore pe semestru	100
3.9. Numărul de puncte de credit	4

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	
4.2. de competențe	Cunoașterea funcționării rețelelor de calculatoare Programare în C++ și Java

5. Condiții (acolo unde este cazul)

5.1. De desfășurare a cursului	• Sala trebuie să fie dotată cu tablă și videoproiector
5.2. De desfășurare a seminarului/laboratorului/proiectului	• Laborator dotat cu calculatoare.

6. Competențele specifice acumulate

Competențe profesionale	• C4. Îmbunătățirea performantelor sistemelor hardware, software si de comunicații • C4.1 Identificarea și descrierea elementelor definitorii ale performantelor sistemelor hardware, software și de comunicații • C4.2 Explicarea interacțiunii factorilor care determina performantele sistemelor hardware, software și de comunicații • C4.3 Aplicarea metodelor si principiilor de baza pentru creșterea performantelor sistemelor hardware, software și de comunicații • C4.4 Alegerea criteriilor si metodelor de evaluare a performantelor sistemelor hardware, software și de comunicații • C4.5 Dezvoltarea de soluții profesionale pentru sisteme hardware, software și de comunicații bazate pe creșterea performantelor • C5. Proiectarea, gestionarea ciclului de viată, integrarea și integritatea sistemelor hardware, software și de comunicații • C5.1 Precizarea criteriilor relevante privind ciclul de viată, calitatea, securitatea și interacțiunea sistemului de calcul cu mediul și cu operatorul uman • C5.2 Utilizarea unor cunoștințe interdisciplinare pentru adaptarea sistemului informatic în raport cu cerințele domeniului de aplicații • C5.3 Utilizarea unor principii și metode de bază pentru asigurarea securității, siguranței și ușurinței în exploatare a sistemelor de calcul • C5.4 Utilizarea adecvată a standardelor de calitate, siguranță și securitate în prelucrarea informațiilor • C5.5 Realizarea unui proiect incluzând identificarea și analiza problemei, proiectarea, dezvoltarea și demonstrând o înțelegere a nevoii de calitate
Competențe transversale	• CT1. Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesie

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1. Obiectivul general al disciplinei	• Prezentarea principiilor de organizare si funcționare a sistemelor distribuite, precum si a celor mai importante componente si mecanisme utilizate in programarea acestora.
7.2. Obiectivele specifice	• Prezentarea metodelor de proiectare verificare și evaluarea performanțelor aplicațiilor distribuite. • Compararea diferitelor tehnologii utilizate în implementarea sistemelor distribuite, studii de caz.

	• Scrierea unor aplicații software formate din mai multe module, implementate folosind limbaje diferite (C++, Java, C# etc.), executându-se pe platforme diferite și sub sisteme de operare diferite. • La proiect se va realiza o platformă intermediară pentru aplicații distribuite implementând diferite servicii ale acestora. Proiectarea și implementarea fiecărei funcționalități se realizează pe echipe de studenți, la final realizându-se o comparare a avantajelor și dezavantajelor fiecărei soluții în parte.

8. Conținuturi

8.1. Curs	Metode de predare	Observații
Capitolul I. Structura sistemelor distribuite Obiectivele generale. Modelul arhitectural global. Structura pe mai multe nivele de abstractizare. Alternative arhitecturale. Structura hardware generală. Tehnologii de interconectare. Protocoale de comunicație de bază	Prelegerea, problematizarea, conversația, explicația	10
Capitolul II. Platforme intermediare Elemente specifice pentru sisteme de operare distribuite. Sincronizare. Tranzacții. Mecanisme de detectare și prevenire a interblocajului în sisteme distribuite. Descentralizarea controlului. Replicare și consistență. Servicii de nume. Sistem de fișiere distribuit. Servicii de distribuire a obiectelor.	Prelegerea, problematizarea, conversația, explicația	10
Capitolul III. . Proiectarea și verificarea aplicațiilor distribuite Determinarea cerințelor și crearea specificațiilor. Crearea modelelor analitice. Prezentarea diferitelor tehnici formale pentru verificarea sistemelor distribuite. Evaluarea performanțelor .	Prelegerea, problematizarea, conversația, explicația	8

Bibliografie

1. Tanenbaum A., Steen M.: Distributed Systems: Principles and Paradigms, Prentice Hall, 2009.

Bibliografie opțională

2. Coulouris G., Dollimore J., Kindberg T.: Distributed Systems Concepts and Design, Ed 4, Pearson Education, 2005.
3. Haller P.: Proiectarea și verificarea aplicațiilor distribuite, MatrixROM, București, 2008.
4. Boian F.M, Ferdean C., Boian R., Dragoș R.: Programare concurentă pe platforme Unix, Windows, Java, Editura Albastră, Cluj, 2002.
5. Tanenbaum A.: Sisteme de operare moderne, Byblos, Bucuresti, 2004.
6. Thomas Erl, "Service-Oriented Architecture (SOA): Concepts, Technology, and Design", Prentice Hall, 2005.

8.2. Seminar		
8.3. Laborator	Metode de predare	Observații
1. Fire de execuție, procese, mecanisme de sincronizare	Rezolvare probleme de programare pe tema propusă.	2
2. Tehnici de accesare a bazelor de date	Rezolvare probleme de programare pe tema propusă.	2

3. Proiectarea aplicației distribuite	Rezolvare probleme de programare pe tema propusă.	2
4. Proiectarea protocoalelor	Rezolvare probleme de programare pe tema propusă.	2
5. Apelul procedurii la distanță - RPC	Rezolvare probleme de programare pe tema propusă.	2
6. Aplicație de gestiune utilizând componente XPCOM	Rezolvare probleme de programare pe tema propusă.	2
7. Aplicație multimedia utilizând componente XPCOM	Rezolvare probleme de programare pe tema propusă.	2
8.4. Proiect	Metode de predare	Observații
Proiect: Proiectarea unei aplicații de gestionare a bibliotecii	projectare	2
Implementarea serverului	programare	4
Implementarea clientului	programare	4
Implementarea serverului de nume	programare	2
Verificarea si evaluarea proiectului	Verificare	2

Bibliografie

1. Tanenbaum A., Steen M.: Distributed Systems: Principles and Paradigms, Prentice Hall, 2009.
2. Coulouris G., Dollimore J., Kindberg T.: Distributed Systems Concepts and Design, Ed 4, Pearson Education, 2005.
3. Haller P.: Proiectarea și verificarea aplicațiilor distribuite, MatrixROM, București, 2008.
4. Boian F.M, Ferdean C., Boian R., Dragoș R.: Programare concurentă pe platforme Unix, Windows, Java, Editura Albastră, Cluj, 2002.
5. Tanenbaum A.: Sisteme de operare moderne, Byblos, Bucuresti, 2004.
6. Thomas Erl, "Service-Oriented Architecture (SOA): Concepts, Technology, and Design", Prentice Hall, 2005.

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

- Disciplina asigură familiarizarea studenților cu principiile sistemelor distribuite și este în concordanță cu tematica de la cursuri similare la alte universități din țară și străinătate.

10. Evaluare

A. Condiții de îndeplinit pentru prezentarea la evaluare:

Susținerea proiectului elaborat și obținerea cel puțin a notei 5 la proiect.

B. Criterii, metode și ponderi în evaluare:

Tip activitate		10.1. Criterii de evaluare	10.2. Metode de evaluare	10.3. Pondere din nota finală
10.4. Curs		Cunoașterea noțiunilor și conceptelor fundamentale din capitolul studiat.	Probă scrisă pe parcursul semestrului	15%
		Cunoașterea noțiunilor și conceptelor fundamentale. Capacitatea de a analiza și de a alege metoda de rezolvare optimă pentru un sistem distribuit.	Probă scrisă.	15%
10.5 .	Seminar			
	Laborator	Însușirea și înțelegerea problematicii tratate la curs și laborator.	Discuții și probe practice în fiecare ședință de laborator. Verificarea pe parcurs a temelor individuale propuse.	0%
	Proiect	Evaluarea modului de realizare a proiectului, respectarea cerințelor formulate.	Prezentarea proiectului	70%
10.6. Standard minim de performanță				
Cunoașterea și înțelegerea noțiunilor de bază din domeniul sistemelor distribuite. Capacitatea de a dezvolta un sistem distribui minimal.				

Semnătura titularului disciplinei:

Șef lucr. dr. ing.Vajda Tamás

Semnătura titularului /rilor de aplicații:

Șef lucr. dr. ing.Vajda Tamás

Data:

16. 09. 2019

Semnătura directorului de departament:

Conf. dr. ing. Domokos József

Semnătura coordonatorului programului de studii:

Ș.l.dr.ing. Szabó László Zsolt