FIȘA DISCIPLINEI

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea „Sapientia" din municipiul Cluj-Napoca
1.2. Facultatea/ DSPP	Facultatea de Științe Tehnice și Umaniste din Târgu Mureș
1.3. Domeniul de studii	Calculatoare și 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				Matematică-Informatică			
2.1 Denumirea disciplinei				Proiectare software			
2.2 Titularul disciplinei / activităților de curs				Conf.dr. Iclănzan Dávid			
2.3 Titularul (ii) activităților de	seminar			-			
	laborator			Conf.dr. Iclănzan Dávid			
	proiect			-			
2.4 Anul de studiu	4	2.5 Semestrul	7	2.6. Tipul de evaluare	EX	2.7 Regimul disciplinei	DO

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	0 /2/ 0
3.4 Total ore din planul de învățământ	56	Din care: 3.5 curs	28	3.6 seminar / laborator / proiect	0 /28 /0

Distribuția fondului de timp:	ore
a) Studiul după manual, suport de curs, bibliografie și notițe	25
b) Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	5
c) Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	11
d) Tutoriat	-
e) Examinări	3
f) Alte activități:	-

3.7 Total ore studiu individual	44
3.8 Total ore pe semestru	100
3.9 Numărul de credite	4

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	
4.2 de competențe	

5. Condiții (acolo unde este cazul)

5.1 De desfășurare a cursului	• Sală de curs cu tablă și videoproiector
5.2 De desfășurare a seminarului/laboratorului/proiectului	• Laborator echipat cu calculatoare cu mediu de programare C++ instalate.

6. Competențele specifice acumulate

Competențe profesionale	• C3. Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor • C3.1 Identificarea unor clase de probleme și metode de rezolvare caracteristice sistemelor informatice • C3.2 Utilizarea de cunoștințe interdisciplinare, a tiparelor de soluții și a uneltelor, efectuarea de experimente și interpretarea rezultatelor lor • C3.3 Aplicarea tiparelor de soluții cu ajutorul uneltelor și metodelor inginerești • C3.4 Evaluarea comparativă, inclusiv experimentală, a alternativelor de rezolvare, pentru optimizarea performanțelor • C3.5 Dezvoltarea și implementarea de soluții informatice pentru probleme concrete
Competențe transversale	•

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	• Dezvoltarea gândirii științifice, familiarizarea viitorului specialist cu proiectarea aplicațiilor orientate pe obiecte, analiza soluțiilor arhitecturale pe diferite nivele de abstractizare.
7.2 Obiectivele specifice	• Prezentarea și analiza stilurilor arhitecturale. • Studiul șabloanelor arhitecturale și de proiectare. • Studiul principiilor de proiectare a claselor si pachetelor.

8. Conținuturi

8.1. Curs	Metode de predare	Observații
Introducere în problematica proiectării aplicațiilor OO.	expunere	2 ore
Elemente avansate de UML (modelarea constrângerilor).	expunere	2 ore
Proiectare arhitecturală, stiluri arhitecturale.	expunere	2 ore
Principiile de proiectare SOLID.	expunere	2 ore
Proiectare OO.	expunere	2 ore
Proiectarea claselor și a pachetelor.	expunere	2 ore
Șabloane de proiectare creaționale.	expunere	4 ore
Șabloane de proiectare comportamentale.	expunere	4 ore
Șabloane de proiectare structurale.	expunere	4 ore
Șabloane de proiectare funcționale.	expunere	2 ore
Metrici de calitate în proiectarea software.	expunere	2 ore

Bibliografie obligatorie

1. Nesteruk, Dmitri. *Design patterns in modern C++: reusable approaches for object-oriented software design*, Apress, 2018.
2. Alexandrescu, Andrei. *Modern C++ design: generic programming and design patterns applied.* Addison-Wesley, 2001.

Bibliografie opțională

3. E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Design Patterns. Addison-Wesley, 1995.
4. Craig Larman, Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development (3rd Edition), Prentice Hall, 2004.
5. Ian Gorton, Essential Software Architecture, Springer, second ed. 2011.
6. Taylor, R., Medvidovic, N., Dashofy, E., Software Architecture: Foundations, Theory, and Practice, 2010, Wiley.

8.2 Seminar – nu este cazul	Metode de predare	Observații
8.3 laborator		
UML, șabloane, tehnici de testare	programare	2 ore
Bazele XML, JSON	programare	2 ore
Principii SOLID	programare	2 ore
Stiluri de proiectare arhitecturale	programare	2 ore
Proiectarea claselor și a pachetelor	programare	2 ore
Șabloane de proiectare creaționale.	programare	4 ore
Șabloane de proiectare comportamentale.	programare	4 ore
Șabloane de proiectare structurale.	programare	4 ore
Șabloane de proiectare funcționale.	programare	4 ore
Verificare finală	programare	2 ore

Bibliografie

1. Nesteruk, Dmitri. *Design patterns in modern C++: reusable approaches for object-oriented software design*. Apress, 2018
2. E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Design Patterns. AddisonWesley, 1995.
3. Craig Larman, Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development (3rd Edition), Prentice Hall, 2004, ISBN: 0131489062

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

- Conținutul disciplinei este în concordanță cu cerințele planului de învățământ, cu conținutul disciplinei predate la alte Universități din țară (Universitatea Babeș-Bolyai, Universitatea Tehnică din Cluj-Napoca) și de peste hotare (Universitatea Tehnică și de Științe Economice din Budapesta)
- Programarea, proiectarea soluțiilor paralele se realizează folosind tehnologii standard în industrie, conținutul cursului fiind adaptat pieței muncii in domeniul ITC.

10. Evaluare

A. Condiții de îndeplinit pentru prezentarea la evaluare:

Efectuarea tuturor lucrărilor de laborator și promovarea testului de evaluare la laborator

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		înțelegerea problemei, capacitate de a proiecta soluții arhitecturale	examen scris cu subiect de teorie și rezolvare de probleme	40%
10.5	Seminar	-	-	-
	Laborator	înțelegerea problemei și rezolvarea problemelor	Verificarea activității de laborator pe parcurs / și la sfârșitul semestrului	10% / 50%
	Proiect	-	-	-

10.6 Standard minim de performanță
1. Cunoașterea principiilor SOLID. 2. Cunoașterea șabloanelor de proiectare creaționale, comportamentale, structurale și funcționale. 3. Capacitatea de a dezvolta soluții arhitecturale pentru specificațiile unor aplicații concrete

Semnătura titularului disciplinei:
Conf.dr. Iclănzan Dávid

Semnătura titularului /rilor de aplicații:
Conf.dr. Iclănzan Dávid

Data:
16. 09. 2019

Semnătura directorului de departament:
Conf. dr. ing. Kátai Zoltán

Semnătura coordonatorului programului de studii:
Ș.l.dr.ing. Szabó László Zsolt