

## FIȘA DISCIPLINEI

### 1. Date despre program

1.1 Instituția de învățământ superior	<b>Universitatea Babeș-Bolyai Cluj-Napoca</b>
1.2 Facultatea	<b>Facultatea de Matematica și Informatică</b>
1.3 Departamentul	<b>Departamentul de Informatică</b>
1.4 Domeniul de studii	Informatică
1.5 Ciclul de studii	Postuniversitar
1.6 Programul de studiu / Calificarea	<b>Program postuniversitar de formare și dezvoltare profesională în Informatică</b>

### 2. Date despre disciplină

2.1 Denumirea disciplinei	<b>Elaborarea lucrării de absolvire</b>						
2.2 Titularul activităților de curs	Lect. dr. Simon Károly						
2.3 Titularul activităților de seminar	Lect. dr. Simon Károly						
2.4 Anul de studiu	<b>2</b>	2.5 Semestrul	<b>3</b>	2.6. Tipul de evaluare	<b>E</b>	2.7 Regimul disciplinei	<b>obligatoriu</b>

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

3.1 Număr de ore pe săptămână	8	Din care: 3.2 curs	0	3.3 seminar/laborator	8
3.4 Total ore din planul de învățământ	80	Din care: 3.5 curs	0	3.6 seminar/laborator	80
Distribuția fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie și notițe					10
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					25
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					90
Tutoriat					40
Examinări					5
Alte activități: .....					-
3.7 Total ore studiu individual					170
3.8 Total ore pe semestru					250
3.9 Numărul de credite					10

### 4. Precondiții (acolo unde este cazul)

4.1 de curriculum	<ul style="list-style-type: none"> <li>Fundamentele Programării și Algoritmica, Programare și Structuri de Date, Programare Orientată Obiect, Metode și Medii avansate de programare, Baze de date, Software Engineering și șabloane de proiectare</li> </ul>
4.2 de competențe	<ul style="list-style-type: none"> <li>Cunoștințe de programare într-un limbaj de programare de nivel înalt, orientat obiect</li> </ul>

### 5. Condiții (acolo unde este cazul)

5.1 De desfășurare a cursului	
5.2 De desfășurare a seminarului/laboratorului	Activitatea trebuie să includă activități de specialitate: programare, testare, analiza și proiectare sisteme software

## 6. Competențele specifice acumulate

<b>Competențe profesionale</b>	<p>C2.1 Identificarea de metodologii adecvate de dezvoltare a sistemelor software</p> <p>C2.2 Identificarea și explicarea mecanismelor adecvate de specificare a sistemelor software</p> <p>C2.3 Utilizarea metodologiilor, mecanismelor de specificare și a mediilor de dezvoltare pentru realizarea aplicațiilor informatice</p>
<b>Competențe transversale</b>	<p><b>CT1</b> Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională</p> <p><b>CT3</b> Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare</p>

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

7.1 Obiectivul general al disciplinei	Realizarea unui sistem informatic, documentarea și prezentarea rezultatelor
7.2 Obiectivele specifice	<ul style="list-style-type: none"> <li>- Analiza și formalizarea unei probleme specifice</li> <li>- Aplicarea conceptelor teoretice învățate pentru realizarea unui sistem software</li> <li>- Documentarea și prezentarea rezultatelor</li> </ul>

## 8. Conținuturi

8.1 Curs	Metode de predare	Observații
8.2 Seminar / laborator	Metode de predare	Observații
1. Specificarea temei (enunțul problemei)	explicare, exemplificare, dezbatere, dialog	
2. Documentare pe platforme de specialitate	explicare, exemplificare, dezbatere, dialog	
3. Planul lucrării (titlu, introducere, conținut)	explicare, exemplificare, dezbatere, dialog	
4. Realizarea proiectului: analiza, design, implementare, testare	explicare, exemplificare, dezbatere, dialog	
5. Documentarea rezultatelor	exemplificare, dialog, evaluare	

6. Elaborarea lucrării de absolvire (pe baza rezultatelor obținute)	exemplificare, dialog, evaluare	
7. Prezentarea proiectului și a lucrării de absolvire	Prezentare, evaluare	
Bibliografie		
1. Eckel, B.: Thinking in Patterns. Problem-Solving Techniques using Java ( <a href="http://www.bruceeckel.com">http://www.bruceeckel.com</a> ). 2. Fowler M.: Patterns of Enterprise Application Architecture, Addison-Wesley, Pearson Education, 2003. 3. Gamma E., Helm R., Johnson R., Vlissides J.: Design Patterns - Elements of Reusable Object-Oriented Software, Addison-Wesley, 1995 4. Alte surse specifice proiectului		

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

Cursul respectă recomandările IEEE și ACM legate de Curricula pentru specializarea Informatică

- Cursul oferă o imagine de ansamblu asupra mai multor domenii din Informatică, oferă studentului o expertiză generală asupra Informaticii.
- Cursul oferă cunoștințe de bază despre lucrul în echipă și integrare în piața muncii

### 10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs			
Seminar/laborator		Prezentarea și evaluarea lucrării	50%
		Evaluarea proiectului	50%
Standard minim de performanță			
<ul style="list-style-type: none"> <li>• Minimum 5 la fiecare probă</li> </ul>			

Data completării

Semnătura titularului de curs

Semnătura titularului de seminar

Lect. dr. Simon Károly

Lect. dr. Simon Károly

Data avizării în departament

Semnătura directorului de departament

.....

Conf. dr. András Szilárd