

## SYLLABUS

### 1. Information regarding the programme

1.1 Higher education institution	<b>Babe Bolyai University</b>
1.2 Faculty	<b>Faculty of Mathematics and Computer Science</b>
1.3 Department	<b>Department of Computer Science</b>
1.4 Field of study	<b>Computer Science</b>
1.5 Study cycle	<b>Bachelor</b>
1.6 Study programme / Qualification	<b>Computer Science</b>

### 2. Information regarding the discipline

2.1 Name of the discipline	<b>Systems for Design and Implementation</b>						
2.2 Course coordinator	<b>Lect. PhD. Grigoreta Cojocar</b>						
2.3 Seminar coordinator	<b>Lect. PhD. Grigoreta Cojocar</b>						
2.4. Year of study	<b>3</b>	2.5 Semester	<b>6</b>	2.6. Type of evaluation	<b>E</b>	2.7 Type of discipline	<b>Compulsory</b>

### 3. Total estimated time (hours/semester of didactic activities)

3.1 Hours per week	4	Of which: 3.2 course	2	3.3 seminar/laboratory	2 lab
3.4 Total hours in the curriculum	48	Of which: 3.5 course	24	3.6 seminar/laboratory	24
Time allotment:					hours
Learning using manual, course support, bibliography, course notes					12
Additional documentation (in libraries, on electronic platforms, field documentation)					20
Preparation for seminars/labs, homework, papers, portfolios and essays					40
Tutorship					10
Evaluations					20
Other activities: .....					-
3.7 Total individual study hours	102				
3.8 Total hours per semester	150				
3.9 Number of ECTS credits	6				

### 4. Prerequisites (if necessary)

4.1. curriculum	<ul style="list-style-type: none"> <li>• Advanced Programming Methods</li> <li>• Databases</li> <li>• Distributed Operating System</li> </ul>
4.2. competencies	<ul style="list-style-type: none"> <li>• Average programming skills in a high level programming language</li> <li>• Basic concepts of databases</li> <li>• Basic concepts of networking</li> </ul>

## 5. Conditions (if necessary)

5.1. for the course	•
5.2. for the seminar /lab activities	• Laboratory with computers; high level programming language environment (.NET and Java.), databases

## 6. Specific competencies acquired

<b>Professional competencies</b>	<ul style="list-style-type: none"> <li>• Knowledge, understanding and use of basic concepts of distributed software</li> <li>• Ability to work independently in order to design and implement a medium distributed application.</li> <li>• Advanced programming skills in high-level languages</li> </ul>
<b>Transversal competencies</b>	<ul style="list-style-type: none"> <li>• Ability to apply object oriented programming to different real life problems</li> <li>• Ability to use different frameworks and technologies for medium size applications</li> </ul>

## 7. Objectives of the discipline (outcome of the acquired competencies)

7.1 General objective of the discipline	<ul style="list-style-type: none"> <li>• Be able to understand distributed software concepts and problems</li> <li>• Improved design and programming skills</li> </ul>
7.2 Specific objective of the discipline	<ul style="list-style-type: none"> <li>• To have a systematic knowledge concerning application development methodologies</li> <li>• To be familiarized with the modern concepts and preoccupations in the field of developing application software</li> <li>• To know the use of computer-aided software development tools</li> </ul>

## 8. Content

8.1 Course	Teaching methods	Remarks
1. UML Meta-Models and Case Tools - structural elements	Exposure: description, explanation, examples, discussion of case studies	
2. UML Meta-Models and Case Tools - behavioral elements	Exposure: description, explanation, examples, discussion of case studies	
3. Model transformations (M2M, M2T)	Exposure: description, explanation, examples, debate, dialogue	
4. REST services	Exposure: description, explanation, examples, discussion of case studies	
5. Server-side Rich Internet Applications	Exposure: description, explanation, examples, proofs	
6. Securing web applications and services	Exposure: description, explanation, examples, proofs, debate, dialogue	
7. Client-side Rich Internet Applications	Exposure: description,	

	explanation, examples, discussion of case studies	
8. Web Services - Document Oriented (Contract-First)	Exposure: description, explanation, examples	
9. Web Services - RPC Based	Exposure: description, explanation, examples, discussion of case studies	
10. Enterprise Application Integration – SOA	Exposure: description, explanation, examples, debate	
11. Enterprise Application Integration - Message Based	Exposure: description, explanation, examples, discussion of case studies	
12. Enterprise Application Integration - Lightweight Approaches	Exposure: description, explanation, examples, discussion of case studies	

#### Bibliography

1. Joseph Albahari and Ben Albahari, C# 4.0 in a Nutshell, Fourth Edition, O'Reilley, 2010.
2. Larman, C.: Applying UML and Design Patterns: An Introduction to OO Analysis and Design and Unified Process, Berlin, Prentice Hall, 2002.
3. Fowler, M., Patterns of Enterprise Application Architecture, Addison-Wesley, 2002.
4. Hohpe, G., Woolf, B., Enterprise integration patterns, Addison-Wesley, 2003.
5. \*\*\*, Microsoft Developer Network, Microsoft Inc., <http://msdn.microsoft.com/>
6. \*\*\*, The Java Tutorial, SUN Microsystems, Inc, 2004. <http://download.oracle.com/javase/tutorial/>
7. Eckel, B., Thinking in Java, 4th edition, Prentice Hall, 2006
8. David Chappell, Introducing SCA, 2007
9. Walls, Craig, Spring in Action, Third Edition, Ed. O'Reilley, 2011.
10. Spring Documentation <http://www.springsource.org>

8.2 Laboratory	Teaching methods	Remarks
W1 Assignment of the application. The students have to design and develop a distributed desktop client-server application.	Dialogue, case studies	
Weeks 2-3: Design of the application using a CASE tool.	Dialogue, case studies, evaluation	
Weeks 4-6: The implementation of the first solution, either in C# or Java.	Dialogue, case studies, evaluation	
Weeks 7-9: The implementation of the second solution, either in C# or Java.	Dialogue, case studies, evaluation	
Weeks 10-11: The implementation of the third solution, having both Java and C# parts.	Dialogue, case studies, evaluation	
Week 12: The extension of the solution, by adding a web part to the application	Dialogue, case studies, evaluation	

#### Bibliography

1. Joseph Albahari and Ben Albahari, C# 4.0 in a Nutshell, Fourth Edition, O'Reilley, 2010.
2. \*\*\*, Microsoft Developer Network, Microsoft Inc., <http://msdn.microsoft.com/>
3. \*\*\*, The Java Tutorial, SUN Microsystems, Inc, 2004. <http://download.oracle.com/javase/tutorial/>
4. Walls, Craig, Spring in Action, Third Edition, Ed. O'Reilley, 2011.
5. Spring Documentation <http://www.springsource.org>

### 9. Corroborating the content of the discipline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

- The course respects the IEEE and ACM Curricula Recommendations for Computer Science studies;
- The content of the course is considered by software companies as important for average designing

and advanced programming skills .

### 10. Evaluation

Type of activity	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Share in the grade (%)
10.4 Course	<ul style="list-style-type: none"><li>To know the basic concepts of developing distributed applications</li><li>To apply the concepts to design and develop a small client-server application</li></ul>	Practical exam	60%
10.5 Seminar/lab activities	- be able to design a medium client-server application using different technologies	-Practical examination -documentation -observation	40%
10.6 Minimum performance standards			
At least grade 5 (from a scale of 1 to 10) at practical exam, and the final grade at least 5.			

Date                      Signature of course coordinator

Signature of seminar coordinator

01.09.2013      Lect. PhD. Grigoreta Cojocar

Lect. PhD. Grigoreta Cojocar

Date of approval

Signature of the head of department

.....

.....