

SYLLABUS

1. Information regarding the programme

1.1 Higher education institution	Babeş-Bolyai University of Cluj-Napoca
1.2 Faculty	Faculty of Mathematics and Computer Science
1.3 Department	Department of Computer Science
1.4 Field of study	Computer Science
1.5 Study cycle	Master
1.6 Study programme / Qualification	High Performance Computing and Big Data Analytics

2. Information regarding the discipline

2.1 Name of the discipline	Agent based software engineering						
2.2 Course coordinator	Prof. PhD Czibula Gabriela						
2.3 Seminar coordinator	Prof. PhD Czibula Gabriela						
2.4. Year of study	1	2.5 Semester	2	2.6. Type of evaluation	E	2.7 Type of discipline	Optional

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

3.1 Hours per week	3	Of which: 3.2 course	2	3.3 seminar/laboratory	1 sem
3.4 Total hours in the curriculum	42	Of which: 3.5 course	28	3.6 seminar/laboratory	14
Time allotment:					hours
Learning using manual, course support, bibliography, course notes					30
Additional documentation (in libraries, on electronic platforms, field documentation)					41
Preparation for seminars/labs, homework, papers, portfolios and essays					42
Tutorship					10
Evaluations					10
Other activities:					-
3.7 Total individual study hours	133				
3.8 Total hours per semester	175				
3.9 Number of ECTS credits	7				

4. Prerequisites (if necessary)

4.1. curriculum	
4.2. competencies	

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 or any Java environment a.s.o.)

6. Specific competencies acquired

Professional competencies	<ul style="list-style-type: none"> Assimilation of mathematical concepts and formal models to understand the methods and components of high performance systems. Capability of developing of high performance programs based on parallel and distributed programming. Analysis, design, and implementation of data analysis systems.
Transversal competencies	<ul style="list-style-type: none"> Ethic and fair behavior, commitment to professional deontology Team work capabilities; able to fulfill different roles Professional communication skills; concise and precise description, both oral and written, of professional results, negotiation abilities. Entrepreneurial skills; working with economical knowledge; continuous learning

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

7.1 General objective of the discipline	<ul style="list-style-type: none"> To present the field of agents as a new research and application domain of Software Engineering and Artificial Intelligence.
7.2 Specific objective of the discipline	<ul style="list-style-type: none"> To introduce the main concepts and methods related to agent oriented software engineering. To present the connection between agents and other programming paradigms. To present the connection between multiagent systems and the distributed artificial intelligence field. To induce the necessity of MAS through the study of relevant industrial and practical applications.

8. Content

8.1 Course	Teaching methods	Remarks
1. Introduction <ul style="list-style-type: none"> Agent based software engineering The concept of agent and intelligent agent Applications 	<ul style="list-style-type: none"> Interactive exposure Explanation Conversation Didactical demonstration 	
2. Agents and intelligent agents (1) <ul style="list-style-type: none"> Definitions, properties, taxonomies Abstract and concrete architectures for intelligent agents Software agents Mobile agents, interface agents 	<ul style="list-style-type: none"> Interactive exposure Explanation Conversation Didactical demonstration 	
3. Agents and intelligent agents (2) <ul style="list-style-type: none"> Application domains Agents and Objects Agents and Expert Systems Agent based development 	<ul style="list-style-type: none"> Interactive exposure Explanation Conversation Didactical demonstration 	
4. Agent based systems (1) <ul style="list-style-type: none"> Design principles of an agent based system 	<ul style="list-style-type: none"> Interactive exposure 	

<ul style="list-style-type: none"> • Conceptual modeling using agents • Examples 	<ul style="list-style-type: none"> • Explanation • Conversation • Didactical demonstration 	
5. Agent based systems (2) <ul style="list-style-type: none"> • Agents in complex software systems • Implementation of the agent function • Examples 	<ul style="list-style-type: none"> • Interactive exposure • Explanation • Conversation • Didactical demonstration 	
6. Multiagent systems and societies of agents <ul style="list-style-type: none"> • Coordination, cooperation, communication - protocols • Negotiation • Communication languages between agents • KQML, FIPA-ACL 	<ul style="list-style-type: none"> • Interactive exposure • Explanation • Conversation • Didactical demonstration 	
7. Applications of agents and MAS (1) <ul style="list-style-type: none"> • Agents in e-business and e-commerce • Agents in e-banking • Agents for Distributed Data Mining 	<ul style="list-style-type: none"> • Interactive exposure • Explanation • Conversation • Didactical demonstration 	
8. Applications of agents and MAS (2) <ul style="list-style-type: none"> • Information agents • Industrial applications of MAS 	<ul style="list-style-type: none"> • Interactive exposure • Explanation • Conversation • Didactical demonstration 	
9. Distributed problem solving <ul style="list-style-type: none"> • Agent based modeling • Advantages of using agents 	<ul style="list-style-type: none"> • Interactive exposure • Explanation • Conversation • Didactical demonstration 	
10. Distributed constraint satisfaction problems <ul style="list-style-type: none"> • The problem definition • The hyperresolution based consistency algorithm • Asynchronous backtracking • Examples 	<ul style="list-style-type: none"> • Interactive exposure • Explanation • Conversation • Didactical demonstration 	
11. Distributed path finding problems <ul style="list-style-type: none"> • Asynchronous dynamic programming • Learning Real Time A* • Bidirectional search algorithm • Real time multiagent search algorithm • Examples 	<ul style="list-style-type: none"> • Interactive exposure • Explanation • Conversation • Didactical demonstration 	
12. Learning in multiagent systems <ul style="list-style-type: none"> • Types of learning • Cooperative learning in multiagent systems • Team learning • Concurrent learning • Application domains for multiagent learning 	<ul style="list-style-type: none"> • Interactive exposure • Explanation • Conversation • Didactical demonstration 	
13. ASE research reports presentation	<ul style="list-style-type: none"> • Interactive exposure • Conversation 	
14. ASE research reports presentation	<ul style="list-style-type: none"> • Interactive exposure 	

	• Conversation	
Bibliography		
<ol style="list-style-type: none"> 1. M. Wooldridge, G. Weiss, and P.Ciancarini, editors: Agent-Oriented Software Engineering II Springer-Verlag Lecture Notes in Computer Science Volume 2222, February 2001. 2. F. Zambonelli, N. R. Jennings, and M. Wooldridge. Developing Multiagent Systems: The Gaia Methodology. In ACM Transactions on Software Engineering Methodology, 12(3):317-370, July 2003. 3. Czibula, G., Sisteme multiagent în Inteligența Artificială Distribuîtă. Arhitecturi și aplicații. Editura RisoPrint, Cluj-Napoca, 2006 4. Weiss, G. (Ed.): Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, MIT Press, 1999 		
8.2 Seminar / laboratory	Teaching methods	Remarks
		The seminar is structured as 2 hours classes every second week
1. Administration of seminars. Survey of the sources of information available on Internet and Intranet	<ul style="list-style-type: none"> • Interactive exposure • Explanation • Conversation 	
2. Survey of the sources of information available on Internet and Intranet; choosing the paper topic and scheduling the presentation.	<ul style="list-style-type: none"> • Documentation • Explanation • Conversation 	
<i>An agent based system (Project 1) will be developed using an open source agent development environment. The second project (Project 2) will be realized from scratch and documented. The software will have to demonstrate the use of multiple agents for some specific task.</i>		
3. Problem definition and specification for Project 2	<ul style="list-style-type: none"> • Lab assignment • Explanation • Conversation 	
4. Comments about the solution (problem analysis) and conceptual modeling of the problem using agents (Project 2). Demonstration of Project 1	<ul style="list-style-type: none"> • Lab assignment • Explanation • Conversation 	
5. Design documentation for Project 2	<ul style="list-style-type: none"> • Lab assignment • Explanation • Conversation 	
6. Design documentation for Project 2	<ul style="list-style-type: none"> • Lab assignment • Explanation • Conversation 	
7. The electronic version of the source code, test files and any other files required to test Project 2. Project 2 demonstration	<ul style="list-style-type: none"> • Lab assignment • Explanation • Conversation 	
Bibliography		
<ol style="list-style-type: none"> 1. M. Wooldridge, G. Weiss, and P.Ciancarini, editors: Agent-Oriented Software Engineering II Springer-Verlag Lecture Notes in Computer Science Volume 2222, February 2001. 2. F. Zambonelli, N. R. Jennings, and M. Wooldridge. Developing Multiagent Systems: The Gaia Methodology. In ACM Transactions on Software Engineering Methodology, 12(3):317-370, July 2003. 3. Czibula, G., Sisteme multiagent în Inteligența Artificială Distribuîtă. Arhitecturi și aplicații. Editura RisoPrint, Cluj-Napoca, 2006 4. Weiss, G. (Ed.): Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, MIT 		

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 content of the discipline is consistent with the similar disciplines from other romanian universities and universities from abroad, as well as with the requirements that potential employers would have in the distributed artificial intelligence field.

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"> A theoretical research report on an agent based topic, based on some recent research papers should be prepared and presented 	Evaluation of the research report (a written paper of about 10 pages and an oral presentation)	20%
	<ul style="list-style-type: none"> The correctness and completeness of the accumulated knowledge. 	Written exam (in the regular session)	40%
	<ul style="list-style-type: none"> Class attendance 	4 unmotivated absences are accepted, but each unmotivated absence other than those specified above are penalised	10%
10.5 Seminar/lab activities	<ul style="list-style-type: none"> A software project developed using an open source agent development environment 	Evaluation of the project (documentation and demonstration)	15%
	<ul style="list-style-type: none"> An agent based system fully implemented, without using existing development environments. 	Evaluation of the project (software implementation, documentation and demonstration)	15%
10.6 Minimum performance standards			
<ul style="list-style-type: none"> Each student has to prove that (s)he acquired an acceptable level of knowledge and understanding of the Distributed Artificial Intelligence domain, that (s)he is capable of stating these knowledge in a coherent form, that (s)he has the ability to establish certain connections and to use the knowledge in solving different problems. Successful passing of the exam is conditioned by the final grade that has to be at least 5. 			

Date

30.04.2014

Date of approval

Signature of course coordinator

Prof. dr. Gabriela Czibula

Signature of seminar coordinator

Prof. dr. Gabriela Czibula

Signature of the head of department

Prof. dr. Bazil Pârv