

## 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 in English</b>

### 2. Information regarding the discipline

2.1 Name of the discipline (en) (ro)	<b>Cloud Application Architecture</b> <b>Arhitectura aplicatiilor cloud</b>						
2.2 Course coordinator	Lect Horea Adrian Grebla						
2.3 Seminar coordinator	Lect Horea Adrian Grebla						
2.4. Year of study	<b>3</b>	2.5 Semester	<b>5</b>	2.6. Type of evaluation	<b>C</b>	2.7 Type of discipline	<b>Optional</b>
2.8 Code of the discipline	<b>MLE5153</b>						

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

3.1 Hours per week	<b>4</b>	Of which: 3.2 course	<b>2</b>	3.3 seminar/laboratory	<b>1 lab</b>
3.4 Total hours in the curriculum	42	Of which: 3.5 course	<b>28</b>	3.6 seminar/laboratory	<b>14</b>
Time allotment:					hours
Learning using manual, course support, bibliography, course notes					<b>10</b>
Additional documentation (in libraries, on electronic platforms, field documentation)					<b>10</b>
Preparation for seminars/labs, homework, papers, portfolios and essays					<b>10</b>
Tutorship					<b>4</b>
Evaluations					<b>8</b>
Other activities: .....					
3.7 Total individual study hours	<b>58</b>				
3.8 Total hours per semester	<b>100</b>				
3.9 Number of ECTS credits	<b>4</b>				



### 4. Prerequisites

4.1. curriculum	<ul style="list-style-type: none"> <li>• Computer Networks</li> <li>• Databases</li> <li>• Web Programming</li> </ul>
4.2. competencies	<ul style="list-style-type: none"> <li>• Good programming skills in at least one programming language (Java, JavaScript, C#, etc).</li> </ul>

## 5. Conditions (if necessary)

5.1. for the course	Course hall with projector
5.2. for the seminar /lab activities	Computers with internet access, a modern browser and a programming language environment

## 6. Specific competencies acquired

	<p><b>CI.5</b> Development of program units and corresponding documentation</p> <p><b>C2.1</b> Identify adequate software systems development methodologies</p> <p><b>C4.3</b> Identify models and methods adequate to real life problem solving</p> <p><b>C6</b> Design and administration of computer networks</p>
	<p><b>CTI</b> Applying organized and efficient work rules, responsible attitude towards scientific/teaching domains in order to obtain a creative exploitation of own potential, while respecting the principles and rules of professional ethics</p> <p><b>CT3</b> Use of effective methods and techniques for learning, information, research and capacity to exploit knowledge, to adapt to a dynamic society and communication in English language</p>

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

7.1 General objective of the discipline	<ul style="list-style-type: none"> <li>To introduce the students to cloud computing concepts, the motivation behind migrating to the cloud and the challenges such a migration entails.</li> <li>Walk through a relatable, real life use case and point out the benefits of using a public cloud provider in most commercial software endeavours.</li> </ul>
7.2 Specific objective of the discipline	<ul style="list-style-type: none"> <li>Become familiar with Amazon Web Services core offerings.</li> <li>Get hands on experience in developing cloud native applications.</li> </ul>

## 8. Content

8.1 Course	Teaching methods	Remarks
<p><b>1.</b> Introduction to the cloud landscape</p> <ul style="list-style-type: none"> <li>Basic concepts, types</li> <li>Motivation</li> <li>Providers</li> <li>Pricing models</li> <li>Layout and topology (regions, availability zones)</li> </ul>	Exposure: description, explanation, examples, debate	
<p><b>2.</b> The IaaS model</p> <ul style="list-style-type: none"> <li>Virtual machines</li> <li>Images</li> <li>Storage mechanisms (block, file)</li> <li>Backups</li> </ul>	Exposure: description, explanation, examples, debate	
<p><b>3.</b> Cloud networking</p> <ul style="list-style-type: none"> <li>Private cloud networks</li> <li>Firewalls and access lists</li> <li>Network address translation</li> </ul>	Exposure: description, explanation, examples, debate	

4. Scalability <ul style="list-style-type: none"> <li>• Vertical vs Horizontal</li> <li>• Load balancing</li> <li>• Auto-scaling</li> </ul>	Exposure: description, explanation, examples, debate	
5. Web capabilities <ul style="list-style-type: none"> <li>• Static website hosting</li> <li>• Content distribution</li> <li>• (Dynamic) Domain name servers</li> </ul>	Exposure: description, explanation, examples, debate	
6. Availability, disaster recovery <ul style="list-style-type: none"> <li>• Resilience</li> <li>• Multi-AZ deployments</li> <li>• Disaster recovery strategies</li> </ul>	Exposure: description, explanation, examples, debate	
7. Infrastructure security <ul style="list-style-type: none"> <li>• Users, roles, permissions</li> </ul>	Exposure: description, explanation, examples, debate	
8. The PaaS model <ul style="list-style-type: none"> <li>• Managed runtimes</li> <li>• Container basics</li> <li>• Docker</li> </ul>	Exposure: description, explanation, examples, debate	
9. Managed databases <ul style="list-style-type: none"> <li>• Relational</li> <li>• Key-value (Redis, S3)</li> <li>• Document (Mongo)</li> </ul>	Exposure: description, explanation, examples, debate	
10. Application security <ul style="list-style-type: none"> <li>• OAuth</li> <li>• Identity providers</li> </ul>	Exposure: description, explanation, examples, debate	
11. Integration services <ul style="list-style-type: none"> <li>• Queues</li> <li>• Pub-sub topics</li> <li>• Email services</li> </ul>	Exposure: description, explanation, examples, debate	
12. Serverless <ul style="list-style-type: none"> <li>• Faas</li> <li>• Specific databases (Dynamo, Aurora)</li> </ul>	Exposure: description, explanation, examples, debate	
13. API Design <ul style="list-style-type: none"> <li>• REST</li> <li>• API Gateways</li> </ul>	Exposure: description, explanation, examples, debate	
14. Recap and closing	Examples, debate	
Bibliography <ol style="list-style-type: none"> <li>1. Thomas Erl, Ricardo Puttini, Zaigham Mahmood - Cloud Computing: Concepts, Technology &amp; Architecture, Prentice Hall, 1st edition, 2013</li> <li>2. Thomas Erl , Robert Cope, Amin Naserpour - Cloud Computing Design Patterns, Prentice Hall, 1st edition, 2015</li> <li>3. Cornelia Davis - Cloud Native Patterns: Designing change-tolerant software, Manning Publications, 1st edition, 2019</li> <li>4. Michael J. Kavis - Architecting the Cloud: Design Decisions for Cloud Computing Service Models, Wiley, 1st edition, 2014</li> <li>5. Kief Morris - Infrastructure as Code: Managing Servers in the Cloud, O'Reilly, 1st edition, 2016</li> <li>6. Christopher Barnatt - A Brief Guide to Cloud Computing, Robinson Press; 1st edition, 2010</li> <li>7. Andrew S. Tanenbaum, Maarten van Steen - Distributed Systems: Principles and Paradigms, Pearson Prentice Hall, 3rd edition, 2017</li> </ol>		
8.2 Seminar / laboratory	Teaching methods	Remarks

<b>1. Introduction to the cloud landscape</b> <ul style="list-style-type: none"> <li>• Regions</li> <li>• AZs</li> <li>• AWS Management Console</li> </ul>	Presentation, Dialogue, Case studies	
<b>2. IaaS basics</b> <ul style="list-style-type: none"> <li>• EC2 instances</li> <li>• User data</li> <li>• Key pairs</li> <li>• AMI</li> </ul>	Presentation, Dialogue, Case studies	
<b>3. Cloud networking</b> <ul style="list-style-type: none"> <li>• VPCs</li> <li>• Security groups</li> <li>• Auto-scaling groups</li> <li>• Load balancers</li> <li>• Website hosting</li> </ul>	Presentation, Dialogue, Case studies	
<b>4. PaaS basics</b> <ul style="list-style-type: none"> <li>• Docker</li> <li>• ECS</li> <li>• RDS</li> </ul>	Presentation, Dialogue, Case studies	
<b>5. Integration services</b> <ul style="list-style-type: none"> <li>• SQS</li> <li>• SNS</li> <li>• SES</li> <li>• Cognito</li> </ul>	Presentation, Dialogue, Case studies	
<b>6. Serverless</b> <ul style="list-style-type: none"> <li>• Lambda</li> <li>• API Gateway</li> </ul>	Presentation, Dialogue, Case studies	
<b>7. Project grading and evaluation</b>	Evaluation	
<b>Bibliography</b> <ol style="list-style-type: none"> <li>1. Andreas Wittig, Michael Wittig - Amazon Web Services in Action, Manning Publications, 1st edition, 2015</li> <li>2. Bert David - AWS: Amazon Web Services Tutorial for Beginners, Independently published, 1st edition, 2018</li> <li>3. AWS Educate - <a href="https://aws.amazon.com/education/awseducate/">https://aws.amazon.com/education/awseducate/</a></li> </ol>		

**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**

<ul style="list-style-type: none"> <li>• Similar courses exist in the studying program of major universities in Europe and abroad.</li> <li>• The software organisations recognize the importance of the concepts discussed during this course for both the development of new applications and migration of legacy applications.</li> </ul>
--

**10. Evaluation**

Type of activity	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Share in the grade (%)
10.4 Course	Know the applied	Colloquium, subject	50%

	technologies taught during the course	presentation	
10.5 Seminar/lab activities	Be able to implement course concepts and presented technologies	Project presentation at the end of the semester	50%
10.6 Minimum performance standards			
At least grade 5 (from a scale of 1 to 10) at both presentation and laboratory project.			

Date

Signature of course coordinator

Signature of seminar coordinator

Date of approval

Signature of the head of department