

SYLLABUS

Cloud Computing Architecture

University year 2025-2026

1. Information regarding the programme

1.1. Higher education institution	Babeş-Bolyai University	
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	Bachelor	
1.6. Study programme/Qualification	Computer Science	
1.7. Form of education	With Frequency	

2. Information regarding the discipline

2.1. Name of the discipline		Cloud Application Architecture					Discipline code		MLE5153		
2.2. Course coordinator					lect. Dr. Ing. Horea Adrian Grebla						
2.3. Seminar coordinator					lect. Dr. Ing. Horea Adrian Grebla						
2.4. Year of study		3	2.5. Semester		5	2.6. Type of evaluation		C	2.7. Discipline regime		DO

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

3.1. Hours per week	5	of which: 3.2 course	2	3.3 seminar/laboratory/project	1L+2P
3.4. Total hours in the curriculum	70	of which: 3.5 course	28	3.6 seminar/laboratory/project	42
Time allotment for individual study (ID) and self-study activities (SA)					hours
Learning using manual, course support, bibliography, course notes (SA)					8
Additional documentation (in libraries, on electronic platforms, field documentation)					8
Preparation for seminars/labs, homework, papers, portfolios and essays					8
Tutorship					4
Evaluations					2
Other activities:					
3.7. Total individual study hours	30				
3.8. Total hours per semester	100				
3.9. Number of ECTS credits	4				

4. Prerequisites (if necessary)

4.1. curriculum	<ul style="list-style-type: none"> Computer Networks Databases Web Programming
4.2. competencies	<ul style="list-style-type: none"> Good programming skills in at least one programming

5. Conditions (if necessary)

5.1. for the course	<ul style="list-style-type: none"> Multimedia projector
5.2. for the seminar /lab activities	<ul style="list-style-type: none"> Computers with internet access

6.1. Specific competencies acquired ¹

¹ One can choose either competences or learning outcomes, or both. If only one option is chosen, the row related to the other option will be deleted, and the kept one will be numbered 6.

Professional/essential competencies	<ul style="list-style-type: none"> advanced programming skills in high-level programming languages development and maintenance of software systems
Transversal competencies	<ul style="list-style-type: none"> application of organized and efficient work rules, of responsible attitudes towards the didactic-scientific field, to bring creative value to own potential, with respect for professional ethics principles and norms use of efficient methods and techniques to learn, inform, research and develop the abilities to bring value to knowledge, to adapt at the requirements of a dynamical society and to communicate efficiently in Romanian language and in an international language

6.2. Learning outcomes

Knowledge	<ul style="list-style-type: none"> The graduate has the ability to develop, design and create new applications, systems or products using best practices of the field. The graduate is able to apply architectural styles, design patterns and best practices in the field to design software applications of high complexity.
Skills	<ul style="list-style-type: none"> The graduate has the necessary skills for computer program design and software systems analysis. The graduate has the necessary knowledge of Internet security and is able to apply this knowledge to validate and maintain a computer network that exposes common services that are securely accessible from the outside.
Responsibility and autonomy:	<ul style="list-style-type: none"> The graduate has the ability to evaluate different architectures and possible solutions to a problem and choose the right one for the specific requirements and constraints of the application to be developed.

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

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

8. Content

8.1 Course	Teaching methods	Remarks
1. Introduction to the cloud landscape <ul style="list-style-type: none"> Basic concepts, types 	Exposure: description, explanation, examples, debate	

<ul style="list-style-type: none"> • Motivation • Providers • Pricing models • Layout and topology (regions, availability zones) 		
2. The IaaS model <ul style="list-style-type: none"> • Virtual machines • Images • Storage mechanisms (block, file) • Backups 	Exposure: description, explanation, examples, debate	
3. Cloud networking <ul style="list-style-type: none"> • Private cloud networks • Firewalls and access lists • Network address translation 	Exposure: description, explanation, examples, debate	
4. Scalability <ul style="list-style-type: none"> • Vertical vs Horizontal • Load balancing • Auto-scaling 	Exposure: description, explanation, examples, debate	
5. Web capabilities <ul style="list-style-type: none"> • Static website hosting • Content distribution • (Dynamic) Domain name servers 	Exposure: description, explanation, examples, debate	
6. Availability, disaster recovery <ul style="list-style-type: none"> • Resilience • Multi-AZ deployments • Disaster recovery strategies 	Exposure: description, explanation, examples, debate	
7. Infrastructure security <ul style="list-style-type: none"> • Users, roles, permissions 	Exposure: description, explanation, examples, debate	
9. The PaaS model <ul style="list-style-type: none"> • Managed runtimes • Container basics • Docker 	Exposure: description, explanation, examples, debate	
10. Managed databases <ul style="list-style-type: none"> • Relational • Key-value (Redis, S3) • Document (Mongo) 	Exposure: description, explanation, examples, debate	
11. Application security <ul style="list-style-type: none"> • OAuth • Identity providers 	Exposure: description, explanation, examples, debate	
12. Integration services <ul style="list-style-type: none"> • Queues • Pub-sub topics • Email services 	Exposure: description, explanation, examples, debate	
13. Serverless <ul style="list-style-type: none"> • Faas • Specific databases (Dynamo, Aurora) 	Exposure: description, explanation, examples, debate	
14. API Design <ul style="list-style-type: none"> • REST • API Gateways 	Exposure: description, explanation, examples, debate	
15. Recap and closing	Exposure: description, explanation, examples, debate	
Bibliography <ol style="list-style-type: none"> 1. Thomas Erl, Ricardo Puttini, Zaigham Mahmood - Cloud Computing: Concepts, Technology & Architecture, Prentice Hall, 1st edition, 2013 2. Thomas Erl, Robert Cope, Amin Naserpour - Cloud Computing Design Patterns, Prentice Hall, 1st edition, 2015 3. Cornelia Davis - Cloud Native Patterns: Designing change-tolerant software, Manning Publications, 1st edition, 2019 4. Michael J. Kavis - Architecting the Cloud: Design Decisions for Cloud Computing Service Models, Wiley, 1st edition, 2014 		

5. Kief Morris - Infrastructure as Code: Managing Servers in the Cloud, O'Reilly, 1st edition, 2016 6. Christopher Barnatt - A Brief Guide to Cloud Computing, Robinson Press; 1st edition, 2010 Andrew S. Tanenbaum, Maarten van Steen - Distributed Systems: Principles and Paradigms, Pearson Prentice Hall, 3rd edition, 2017		
8.2 Seminar / laboratory	Teaching methods	Remarks
1. Introduction to the cloud landscape <ul style="list-style-type: none"> • Regions • Azs • AWS Management Console 	Presentation, Dialogue, Case studies	
2. IaaS basics <ul style="list-style-type: none"> • EC2 instances • User data • Key pairs • AMI 	Presentation, Dialogue, Case studies	
3. Cloud networking <ul style="list-style-type: none"> • VPCs • Security groups • Auto-scaling groups • Load balancers • Website hosting 	Presentation, Dialogue, Case studies	
4. PaaS basics <ul style="list-style-type: none"> • Docker • ECS • RDS 	Presentation, Dialogue, Case studies	
5. Integration services <ul style="list-style-type: none"> • SQS • SNS • SES • Cognito 	Presentation, Dialogue, Case studies	
6. Serverless <ul style="list-style-type: none"> • Lambda • API Gateway 	Presentation, Dialogue, Case studies	
7. Project grading and evaluation	Presentation, Dialogue, Case studies	
Bibliography <ol style="list-style-type: none"> 1. Andreas Wittig, Michael Wittig - Amazon Web Services in Action, Manning Publications, 1st edition, 2015 2. Bert David - AWS: Amazon Web Services Tutorial for Beginners, Independently published, 1st edition, 2018 3. AWS Educate - https://aws.amazon.com/education/awseducate/ 		

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

- Similar courses exist in the studying program of major universities in Europe and abroad.
- The software organisations recognize the importance of the concepts discussed during this course

10. Evaluation

Activity type	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Percentage of final grade
10.4 Course	Know the applied technologies taught during the course	Colloquium	50%
10.5 Seminar/laboratory	Be able to implement course concepts and presented technologies	Test	50%
10.6 Minimum standard of performance			
<ul style="list-style-type: none"> • At least grade 5 (from a scale of 1 to 10) at both presentation and laboratory project. 			

11. Labels ODD (Sustainable Development Goals)²

Not applicable.

Date:

...

Signature of course coordinator

lect. Dr. Ing. Horea Adrian Grebla

Signature of seminar coordinator

lect. Dr. Ing. Horea Adrian Grebla

Date of approval:

...

Signature of the head of department

Assoc.prof.phd. Adrian STERCA

² Keep only the labels that, according to the [Procedure for applying ODD labels in the academic process](#), suit the discipline and delete the others, including the general one for *Sustainable Development* – if not applicable. If no label describes the discipline, delete them all and write „*Not applicable.*”.