

Subject card

Subject name and code	Distributed Systems and Cloud Computing, PG_00203645						
Field of study	Informatics						
Date of commencement of studies	October 2026		Academic year of realisation of subject		2027/2028		
Education level	Master's studies		Subject group		Obligatory subject group in the field of study Subject group related to scientific research in the field of study		
Mode of study	full-time studies		Mode of delivery		at the university		
Year of study	2		Language of instruction		Polish		
Semester of study	3		ECTS credits		5.0		
Learning profile	academic		Assessment form		credit		
Conducting unit							
Name and surname of lecturer (lecturers)	Subject supervisor		dr Jakub Neumann				
	Teachers						
Lesson types	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	30.0	0.0	30.0	0.0	0.0	60
	E-learning hours included: 0.0						
Learning activity and number of study hours	Learning activity	Participation in didactic classes included in study plan		Participation in consultation hours		Self-study	SUM
	Number of study hours	60		0.0		65.0	125
Subject objectives	The aim of the course is to provide students with an in-depth understanding of designing, deploying and operating distributed systems in a cloud computing environment. Emphasis on containerization, container orchestration (Kubernetes) as the core mechanism for running distributed services, infrastructure automation in the Infrastructure as Code model, and monitoring/ensuring availability, reliability and security of services.						
Learning outcomes	Course outcome		Subject outcome		Method of verification		
	[INFMU2_W06] knows and understands in depth the concepts of functioning of modern IT systems, understands security threats occurring in networked and distributed systems, knows modern models of data processing and storage		Knows and understands the architecture of distributed cloud systems — processing models, containerization and container orchestration, and security threats of distributed services.		[SW2] presentation/project/paper/report		
	[INFMU2_W09] knows and understands formal models of data representation and structures (e.g. graphs, relational and non-relational models), knows the applications of mathematical models and data structures in solving practical problems, understands the connections between theory (e.g. graphs) and practical data processing systems (e.g. databases, Big Data systems)		Knows and understands models representing the state and configuration of a distributed system (declarative model, dependency graph of infrastructure resources) and the link between formal infrastructure description and its cloud deployment.		[SW2] presentation/project/paper/report		
	[INFMU2_U11] is able to ensure the availability, reliability and security of IT services, design and apply basic security mechanisms		Is able to deploy, configure and maintain a distributed service in a container orchestration environment, ensuring scalability, availability and fault tolerance.		[SU2] presentation/project/paper/report		

Subject contents	1. Distributed systems and cloud computing — service models (IaaS/PaaS/SaaS), deployment models, core properties (scalability, availability, consistency, fault tolerance). 2. Containerization (review) — images and containers, image registries, networks and volumes, composition of multi-container services. 3. Container orchestration in Kubernetes — cluster architecture (control plane and worker nodes), core objects (Pod, ReplicaSet, Deployment, Service). 4. Configuration and data in Kubernetes — ConfigMap, Secret, persistent volumes, application state management. 5. Networking and service exposure in Kubernetes — Service, Ingress, load balancing, service discovery. 6. Scaling and reliability — manual and automatic scaling, health checks (liveness/readiness), deployment and rollback strategies. 7. Application packaging and deployment — version and configuration-template management, multi-tier environments. 8. Infrastructure as Code with Terraform — declarative description of infrastructure, resource dependency graph, plan and state management, module reuse. 9. Monitoring and observability — collecting metrics, logs and traces, visualization, defining alerts and availability thresholds (SLI/SLO). 10. Security of distributed services — authentication and authorization, secret management, isolation, access control (RBAC), basic security mechanisms. 11. Supplementary topics (optional) — CI/CD automation, asynchronous communication and message queues, service mesh.								
Prerequisites and co-requisites	None								
Assessment methods and criteria	<table><tr><td>Subject passing criteria</td><td>Passing threshold</td><td>Percentage of the final grade</td></tr><tr><td>project</td><td>51.0%</td><td>100.0%</td></tr></table>	Subject passing criteria	Passing threshold	Percentage of the final grade	project	51.0%	100.0%		
Subject passing criteria	Passing threshold	Percentage of the final grade							
project	51.0%	100.0%							
Recommended reading	Basic literature	- Kubernetes technical documentation: https://kubernetes.io/docs/ - Docker technical documentation: https://docs.docker.com/ - Terraform technical documentation: https://developer.hashicorp.com/terraform/docs							
	Supplementary literature	- M. Lukša, *Kubernetes in Action*, Manning. - B. Burns, J. Beda, K. Hightower, *Kubernetes: Up and Running*, O'Reilly. - Y. Brikman, *Terraform: Up & Running*, O'Reilly. - Documentation of metric monitoring and visualization systems (Prometheus, Grafana).							
	eResources addresses								
Example issues/ example questions/ tasks being completed	- Compare the IaaS/PaaS/SaaS cloud service models and give application examples. - Describe the architecture of a Kubernetes cluster and the role of each control-plane component. - Deploy a multi-container application in Kubernetes with Ingress exposure and autoscaling. - Define infrastructure in the Infrastructure as Code model and explain the role of the dependency graph and state file. - Configure metric collection and an alert for a chosen service availability indicator (SLO). - Discuss the security mechanisms of distributed services (RBAC, secret management, isolation).								
Work placement	Not applicable								

Document generated electronically. Does not require a seal or signature.