

Karta przedmiotu

Nazwa i kod przedmiotu	Rozproszone systemy o wysokiej dostępności, PG_00063920						
Kierunek studiów	Informatyka						
Data rozpoczęcia studiów	luty 2025 r.		Rok akademicki realizacji przedmiotu		2024/2025		
Poziom kształcenia	II stopnia		Grupa zajęć		Grupa zajęć fakultatywnych Grupa zajęć specjalnościowych Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademicki		
Forma studiów	stacjonarne		Sposób realizacji		na uczelni		
Rok studiów	1		Język wykładowy		polski		
Semestr studiów	1		Liczba punktów ECTS		3.0		
Profil kształcenia	ogólnoakademicki		Forma zaliczenia		zaliczenie		
Jednostka prowadząca	Wydziały Politechniki Gdańskiej -> Wydział Elektroniki, Telekomunikacji i Informatyki -> Katedra Architektury Systemów Komputerowych						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot		dr inż. Andrzej Sobecki				
	Prowadzący zajęcia z przedmiotu		dr inż. Andrzej Sobecki				
Formy zajęć	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	15.0	0.0	0.0	15.0	0.0	30
	W tym liczba godzin zajęć na odległość: 0.0						
	Adresy kursu na platformie eNauczanie: Moodle ID: 45379 Rozproszone systemy o wysokiej dostępności https://enauczanie.pg.edu.pl/moodle/course/view.php?id=45379						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów		Udział w konsultacjach		Praca własna studenta	RAZEM
	Liczba godzin pracy studenta	30		6.0		39.0	75
Cel przedmiotu	Celem przedmiotu jest wprowadzenie studentów w tematykę wytwarzania aplikacji rozproszonych, a także systemów gromadzenia i przetwarzania danych w sposób rozproszony. Ponadto w ramach przedmiotu studenci zostaną zaznajomieni z platformami wdrożeniowymi stosowanymi w przemyśle, których zadaniem jest zarządzanie zbiorami rozproszonych komponentów aplikacyjnych.						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[K7_U12] potrafi w pogłębionym stopniu analizować działanie elementów, układów i systemów związanych z kierunkiem studiów oraz mierzyć ich parametry i badać charakterystyki techniczne, a także planować i przeprowadzać eksperymenty związane z kierunkiem studiów, w tym symulacje komputerowe, interpretować uzyskane wyniki i wyciągać wnioski	Student potrafi zaprojektować system w architekturze mikroserwisowej.	[SU1] Assessment of task fulfilment
	[K7_W11] zna i rozumie w pogłębionym stopniu ogólne zasady tworzenia i rozwoju form indywidualnej przedsiębiorczości oraz ekonomiczne, prawne i inne uwarunkowania różnych rodzajów działań związanych z nadaną kwalifikacją, w tym zasady ochrony własności przemysłowej i prawa autorskiego	Student rozumie korzyści ekonomiczne ze stosowania architektury mikroserwisowej w odniesieniu do tempa rozwoju systemu i kosztów utrzymania dużych systemów monolitycznych.	[SW1] Assessment of factual knowledge
	[K7_W10] zna i rozumie w pogłębionym stopniu podstawowe procesy zachodzące w cyklu życia urządzeń, obiektów i systemów technicznych oraz metody wspomagania procesów i funkcji, specyficzne dla kierunku studiów	Student potrafi przygotować proces CI/CD dla systemu utworzonego w architekturze mikroserwisowej.	[SW1] Assessment of factual knowledge
	[K7_W03] zna i rozumie w pogłębionym stopniu budowę i zasady działania komponentów i systemów związanych z kierunkiem studiów, w tym teorie, metody i złożone zależności między nimi oraz wybrane zagadnienia szczegółowe – właściwe dla programu kształcenia	Student zna i opisuje różne architektury wytwarzania aplikacji. Zna różnice, zalety i wady ze stosowania monolitycznych architektur warstwowych i docelowych architektur rozproszonych.	[SW1] Assessment of factual knowledge
Treści przedmiotu	Treści przedmiotu - wykład Skalowalność aplikacji, deployment; Architektury aplikacji rozproszonych (monolit -> mikro usług (CQRS/Event Sourcing/Saga); Konteneryzacja usług -> docker, docker-compose, docker swarm, kubernetes; Wdrażaniu i utrzymaniu aplikacji rozproszonej -> monitorowanie (klastry/chmury obliczeniowe OpenStack / AWS) Monitorowanie -> Sentry / Jaeger / Prometheus + Grafana / Load balancery / Systemy kolejkowe; Narzędzia do testowania obciążenia Locust.io / Jmeter; Rozproszone systemy plikowe HDFS (Hive) / IPFS; Rozproszone bazy danych (Hbase / Neo4j, ArangoDB) Blockchain -> Bitcoin / Ethereum / Stellar / GRP (grafowa); Rozproszone środowisko obliczeniowe (Apache Spark/ YARN -> JupyterLab -> PySparku -> .net context submit).		
	Treści przedmiotu - projekt Realizacja systemu w architekturze mikroserwisowej, z wykorzystaniem wzorców Saga, CQRS, Event Sourcing. Projekt docelowo wdrażany na klastrze serwerów.		
Wymagania wstępne i dodatkowe			
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	Projekt	50.0%	50.0%
	Kolokwium końcowe	50.0%	50.0%

Zalecana lista lektur	Podstawowa lista lektur	1. Cloud Native DevOps with Kubernetes, John Arundel, Justin Domingus2. Kubernetes Patterns: Reusable Elements for Designing Cloud-Native Applications, Bilgin Ibryam, Roland Huß3. KUBERNETES: A Simple Guide to Master Kubernetes for Beginners and Advanced Users (2020 Edition), Brian Docker4. Hands-On Docker for Microservices with Python: Design, deploy, and operate a complex system with multiple microservices using Docker and Kubernetes, Jaime Buelta5. gRPC: Up and Running: Building Cloud Native Applications with Go and Java for Docker and Kubernetes, Kasun Indrasiri, Danesh Kuruppu6. The Kubernetes Book, Nigel Poulton7. Hands-On Microservices with C# 8 and .NET Core 3: Refactor you monolith architecture into microservices using Azure, 3rd Edition, Gaurav Aroraa, Ed Price8. Pro ASP.NET Core 3: Develop Cloud-Ready Web Applications Using MVC, Blazor, and Razor Pages, Adam Freeman9. Practical Microservices Architectural Patterns - Event-Based Java Microservices with Spring Boot and Spring Cloud, Binildas Christudas10. Monolith to Microservices: Evolutionary Patterns to Transform Your Monolith, Sam Newman11. Practical Microservices: Build Event-Driven Architectures with Event Sourcing and CQRS, Ethan Garofolo12. Architecting Modern Data Platforms, Jan Kuniġk, Ian Buss, Paul Wilkinson & Lars George13. Advanced Analytics with Spark, Sandy Ryza, Uri Laserson, Sean Owen & Josh Wills14. Big Data Analytics with Hadoop 3, Sridhar Alla,15. Modern Big Data Processing with Hadoop, V. Naresh Kumar Prashant Shindgikar
	Uzupełniająca lista lektur	6. The Kubernetes Book, Nigel Poulton 7. Hands-On Microservices with C# 8 and .NET Core 3: Refactor you monolith architecture into microservices using Azure, 3rd Edition, Gaurav Aroraa, Ed Price 8. Pro ASP.NET Core 3: Develop Cloud-Ready Web Applications Using MVC, Blazor, and Razor Pages, Adam Freeman
	Adresy eZasobów	
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania	1. Wyjaśnij czym jest CNCF 2. Czym jest Infrastrcture as a Code 3. RDD vs DataFrame 4. HDFS vs IPFS 5. pySpark vs python 6. Rola Yarna	
Zajęcia praktyczne w ramach przedmiotu	Nie dotyczy	

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.