

# AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

## KARTA OPISU PRZEDMIOTU

<b>Wydział</b>		<b>Informatyki</b>	
<b>Kierunek</b>		<b>Informatyka</b>	
<b>Specjalność</b>		<b>AI DevOps Engineering</b>	
<b>Semestr</b>	<b>VI</b>	<b>Program studiów, dla którego obowiązuje sylabus</b>	<b>2025/2026</b>
<b>Stopień studiów</b>	<b>I</b>		

Nazwa przedmiotu	Projektowanie i zarządzanie systemami DevOps/GitOps			
Kod przedmiotu	PIZSDG			
Łączna liczba godzin	18	Tryb	stacjonarny	niestacjonarny
Profil kształcenia	Ogólnoakademicki (A)		Praktyczny (P)	
Forma zajęć	laboratorium			
Język przedmiotu	polski			
Liczba punktów ECTS	2			

<b>Prowadzący zajęcia</b>	
<b>Forma prowadzonych zajęć</b>	<b>Laboratorium</b>
<b>Wymiar zajęć</b>	<b>18 h</b>
<b>Stopień (tytuł) naukowy</b>	
<b>Imię</b>	
<b>Nazwisko</b>	

<b>Wymagania wstępne</b>	Ukończenie kursu Konteneryzacja i orkiestracja danych.
<b>Założenia i cele przedmiotu</b>	Przedmiot koncentruje się na zaawansowanym wykorzystaniu technologii Docker i Kubernetes, w tym operatorów, CRD, Helm Charts, zarządzaniu siecią (Istio), pamięcią (Persistent Volumes), bezpieczeństwem (RBAC), monitoringiem, logowaniem oraz na automatyzacji zarządzania klastrami w oparciu o GitOps. Celem jest nabycie kompleksowej wiedzy i umiejętności w zakresie projektowania, wdrażania i utrzymania środowisk kontenerowych na dużą skalę.
<b>Metody dydaktyczne</b>	<ol style="list-style-type: none"> <li>1. Prezentacje multimedialne.</li> <li>2. Pokazy przykładowych rozwiązań problemów.</li> <li>3. Rozwiązywanie zadań praktycznych.</li> </ol>

<b>Efekty uczenia się (odniesienie do charakterystyk poziomów Polskiej Ramy Kwalifikacji)</b>		<b>Odniesienie do efektów dla kierunku</b>	<b>Odniesienie do efektów uczenia się wg Polskiej Ramy Kwalifikacji</b>
<b>WIEDZA</b> – absolwent zna i rozumie:	W01. Zaawansowane funkcje Kubernetes (operatorzy, CRD) oraz rolę Helm Charts w zarządzaniu aplikacjami. W02. Zasady konfiguracji sieci w Kubernetes, w tym CNI, Service Mesh (Istio) i Ingress.	K_W14 K_W16 K_W18 K_W23	P6S_WG P6S_WG_INŻ

## AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

	<p>W03. Metody zarządzania pamięcią w Kubernetes (Persistent Volumes, Storage Classes).</p> <p>W04. Zasady bezpieczeństwa w Kubernetes, takie jak RBAC i Network Policies, oraz sposoby zabezpieczania klastrów.</p> <p>W05. Metody monitoringu, logowania i zasady GitOps (Argo CD, Flux) w zarządzaniu klastrami.</p>		
UMIEJĘTNOŚCI – absolwent potrafi:	<p>U01. Wdrażać i zarządzać aplikacjami w Kubernetes z wykorzystaniem operatorów, CRD i Helm Charts.</p> <p>U02. Konfigurować sieć w Kubernetes, w tym wdrażać Service Mesh (Istio) i definiować reguły Ingress.</p> <p>U03. Zarządzać zasobami pamięci w klastrze Kubernetes, definiując Persistent Volumes i Storage Classes.</p> <p>U04. Implementować polityki bezpieczeństwa (RBAC, Network Policies) oraz zabezpieczać klaster Kubernetes.</p> <p>U05. Wdrożyć monitoring (Prometheus, Grafana), logowanie (EFK) oraz stosować GitOps (Argo CD, Flux) do automatyzacji zarządzania klastrem.</p>	<p>K_U01 K_U02 K_U03 K_U04 K_U12 K_U18 K_U24 K_U25</p>	<p>P6S_UW P6S_UW_INŻ P6S_UO P6S_KK P6S_UK</p>
KOMPETENCJE SPOŁECZNE – absolwent jest gotów do	<p>K01. Pracy w zespole, przyjmując w nim różne role.</p> <p>K02. Krytycznej oceny możliwości urządzeń, oprogramowania i systemów dostępnych na rynku IT.</p> <p>K03. Ciągłego samokształcenia się w celu dostosowywania się do dynamicznie zmieniających się technologii.</p>	<p>K_K04 K_K05 K_K06</p>	<p>P6S_UO P6S_KR P6S_KK</p>

Lp.	Tematyka zajęć	Liczba godzin
<b>Forma zajęć – laboratorium</b>		
1	Wprowadzenie do DevOps i GitOps. Filozofia, kultura, porównanie koncepcji.	1
2	Kontrola wersji z Git. Zaawansowane techniki, workflow GitFlow.	2
3	Narzędzia GitOps: Flux, Argo CD – implementacja i zastosowanie	2
4	Continuous Integration/Continuous Deployment (CI/CD). Narzędzia (Jenkins, GitLab CI), pipeline'y, praktyki	4
5	Infrastruktura jako kod (IaC). Terraform, Ansible – podstawy i zastosowania	4

## AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

6	Monitoring i logowanie. Prometheus, Grafana, ELK Stack	2
7	Bezpieczeństwo w DevOps. DevSecOps, praktyki bezpieczeństwa, skanowanie kodu. Zaliczenie.	3

<b>Forma i warunki zaliczenia przedmiotu</b>	Wykonanie projektów. Częstkowe prezentacje, zdawanie raportów, obrona projektów.	
<b>Metody weryfikacji efektów uczenia się</b>		<b>Nr efektu uczenia się z sylabusu</b>
	Ocena projektów i częściowych prezentacji.	W01-W05, U01-U05, K01-K03

<b>Literatura podstawowa</b>	<ol style="list-style-type: none"> <li>1. S. Kane, K. Matthias, <i>Docker. Praktyczne zastosowania</i>, Helion, Gliwice 2019.</li> <li>2. B. Burns, J. Beda, K. Hightower, <i>Kubernetes. Tworzenie niezawodnych systemów rozproszonych</i>, Helion, Gliwice 2020.</li> </ol>
<b>Literatura uzupełniająca</b>	<ol style="list-style-type: none"> <li>1. M. Krief, <i>DevOps w praktyce. Wdrażanie narzędzi Terraform, Azure DevOps, Kubernetes i Jenkins</i>, Helion, Gliwice 2023.</li> </ol>

Nakład pracy studenta	
	Liczba godzin
Zajęcia dydaktyczne	18
Przygotowanie się do zajęć	9
Studiowanie literatury	9
Udział w konsultacjach	2
Przygotowanie projektu / eseju / prezentacji itp.	22
Przygotowanie się do egzaminu / zaliczenia	-
Inne	-
<b>ŁĄCZNY nakład pracy studenta w godz.</b>	<b>60</b>
<b>Liczba punktów ECTS</b>	<b>2</b>