

## KARTA KURSU

<b>Nazwa</b>	<b>Technologie DevOps</b>
<b>Nazwa w j. ang.</b>	<b>DevOps Technologies</b>

Koordinator	Mgr Miłosz Borowiecki	Zespół dydaktyczny
Punktacja ECTS*	5	Mgr Miłosz Borowiecki Mgr Katarzyna Marczak

### Opis kursu (cele kształcenia)

Celem przedmiotu jest wprowadzenie studentów w podstawowe zagadnienia i narzędzia DevOps wykorzystywane we współczesnym procesie wytwarzania oprogramowania. Studenci poznają ideę ciągłej integracji, ciągłego dostarczania, automatyzacji i monitorowania systemów.  
Kurs prowadzony jest w języku polskim.

### Warunki wstępne

Wiedza	<b>Podstawy logiki</b> <b>Wiedza</b> <ul style="list-style-type: none"> <li>podstawy systemów operacyjnych (Linux/Windows),</li> <li>podstawy programowania i sieci komputerowych.</li> </ul>
Umiejętności	<b>Umiejętności</b> <ul style="list-style-type: none"> <li>obsługa systemu kontroli wersji Git (w podstawowym zakresie),</li> <li>umiejętność pracy w terminalu.</li> </ul>
Kursy	

### Efekty uczenia się

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Wiedza	W01: rozumie podstawowe pojęcia i ideę DevOps. W02: zna popularne narzędzia wykorzystywane w procesach CI/CD. W03: ma wiedzę o podstawach konteneryzacji i orkiestracji.	K_W09 K_W12

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Umiejętności	U01: potrafi skonfigurować prosty pipeline CI/CD. U02: umie przygotować i uruchomić kontener Dockera. U03: wykorzystuje system kontroli wersji Git w pracy zespołowej. U04: wdraża aplikację demo z wykorzystaniem narzędzi DevOps.	K_U08 K_U13

Kompetencje społeczne	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
	K01: rozumie znaczenie kultury współpracy i odpowiedzialności w zespole DevOps. K02: potrafi pracować w grupie przy realizacji projektu DevOps.	K_K02 K_K03

### Studia stacjonarne

Organizacja											
Forma zajęć	Wykład (W)	Ćwiczenia w grupach									
		A		K		L		S		P	E
Liczba godzin	10					30					

### Studia niestacjonarne

Organizacja											
Forma zajęć	Wykład (W)	Ćwiczenia w grupach									
		A		K		L		S		P	E
Liczba godzin	6					20					

### Opis metod prowadzenia zajęć

- wykład interaktywny,
- ćwiczenia praktyczne w laboratorium,
- projekt grupowy (przygotowanie pipeline'u CI/CD i wdrożenie aplikacji demo).

### Formy sprawdzania efektów uczenia się

	E – learning	Gry dydaktyczne	Ćwiczenia w szkole	Zajęcia terenowe	Praca laboratoryjna	Projekt indywidualny	Projekt grupowy	Udział w dyskusji	Referat	Praca pisemna (esej)	Egzamin ustny	Egzamin pisemny	Inne
W01	x				x	x		x					
W02					x	x		x					
W03					x	x		x					
W04					x	x		x					
U01					x	x		x					
U02					x	x		x					
U03					x	x		x					
U04					x	x		x					
U05					x	x		x					
U06					x	x		x					
K01							x						

Kryteria oceny	<p>Ocenę dobrą i bardzo dobrą uzyska student, który:</p> <ul style="list-style-type: none"> <li>• potrafi samodzielnie skonfigurować pipeline CI/CD,</li> <li>• wdroży aplikację z użyciem Dockera,</li> <li>• zna podstawowe narzędzia monitorowania i automatyzacji,</li> <li>• potrafi pracować zespołowo zgodnie z zasadami kultury DevOps.</li> </ul>
----------------	--

Uwagi	
-------	--

#### Treści merytoryczne (wykaz tematów)

<ul style="list-style-type: none"> <li>• Wprowadzenie do DevOps – idea, geneza, kultura współpracy</li> <li>• Git i systemy kontroli wersji</li> <li>• CI/CD – podstawowe pojęcia i pipeline'y</li> <li>• Praca z GitHub Actions / GitLab CI</li> <li>• Konteneryzacja – Docker, docker-compose</li> <li>• Wdrożenie aplikacji demo</li> <li>• Podstawy monitorowania i logowania</li> <li>• Przykłady narzędzi: Prometheus, Grafana</li> <li>• DevOps w praktyce – mini projekt zespołowy</li> </ul>
---

#### Wykaz literatury podstawowej

<ol style="list-style-type: none"> <li>1. Gene Kim, Jez Humble, Patrick Debois, John Willis – <i>The DevOps Handbook</i></li> <li>2. Kief Morris – <i>Infrastructure as Code</i></li> <li>3. Dokumentacje narzędzi: Git, Docker, GitHub/GitLab CI</li> </ol>
--

#### Wykaz literatury uzupełniającej

<ol style="list-style-type: none"> <li>1. Jez Humble, David Farley – <i>Continuous Delivery</i></li> <li>2. Dokumentacja projektów open source (Prometheus, Grafana, Kubernetes – w przeglądzie)</li> </ol>
---

#### Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) – **studia stacjonarne**

liczba godzin w kontakcie z prowadzącymi	Wykład	10
	Konwersatorium (ćwiczenia, laboratorium itd.)	30
	Pozostałe godziny kontaktu studenta z prowadzącym	10
liczba godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	20
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	10
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	20
	Przygotowanie do egzaminu/zaliczenia	25
Ogółem bilans czasu pracy		125
Liczba punktów ECTS w zależności od przyjętego przelicznika		5

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) – **studia niestacjonarne**

liczba godzin w kontakcie z prowadzącymi	Wykład	6
	Konwersatorium (ćwiczenia, laboratorium itd.)	20
	Pozostałe godziny kontaktu studenta z prowadzącym	15
liczba godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	19
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	20
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	20
	Przygotowanie do egzaminu/zaliczenia	25
Ogółem bilans czasu pracy		125
Liczba punktów ECTS w zależności od przyjętego przelicznika		5