



SYLABUS PRZEDMIOTU

Nazwa przedmiotu:	Technologie DevOps
Kod przedmiotu:	DEV
Kierunek / Profil:	Informatyka / praktyczny
Tryb studiów:	stacjonarny
Rok / Semestr:	3 / 6
Charakter:	obowiązkowy
Odpowiedzialny:	mmiotk@pjwstk.edu.pl
Wersja z dnia:	19.02.2026

1. Godziny zajęć i punkty ECTS

Wykłady	Ćwiczenia	Laboratoria	Z prowadzącym	Praca własna	Łącznie	ECTS
30 h	—	30 h	60 h	65 h	125 h	5

2. Forma zajęć

Forma zajęć	Sposób zaliczenia
Laboratorium	Zaliczenie z oceną
Wykład	Egzamin

3. Cel dydaktyczny

Celem przedmiotu jest zapoznanie się z techniką konteneryzacji na podstawie narzędzia Docker oraz Kubernetes. Efektem przedmiotu ma być możliwość utworzenia aplikacji typu REST API, która będzie spakowana za pomocą wyżej wymienionych narzędzi.

4. Przedmioty wprowadzające

Przedmiot	Wymagane zagadnienia
Użytkowanie komputerów Technologie internetu	Systemy operacyjne Umiejętność posługiwania się emulatorem terminala w systemie operacyjnym
Znajomość struktur plików i katalogów w systemie operacyjnym	Znajomość pojęcia procesu w systemie operacyjnym
Znajomość protokołu HTTP	—

5. Treści programowe

1. Wprowadzenie do idei konteneryzacji
2. Praca oraz monitorowanie kontenerów
3. Tworzenie obrazów do użycia w kontenerze. Konteneryzacja istniejących aplikacji.
4. Wolumeny oraz rejestr w kontenerze
5. Tworzenie obrazów za pomocą technik multietapowych
6. Konfiguracja zasobów oraz sieci w kontenerach
7. Debugowanie kontenerów
8. Zarządzanie kontenerami na przykładzie narzędzia Docker Compose
9. Architektura konteneryzacji aplikacji
10. Pojęcie Continuous Integration
11. Narzędzie Jenkins jako przykład narzędzia do używania CI/CD
12. Zasada ciągłej interakcji (CI)
13. Proces wdrażania aplikacji za pomocą narzędzia Ansible
14. Proces monitorowania aplikacji w cyklu produkcyjnym
15. Uruchomienie kontenerowej aplikacji w chmurze

6. Efekty kształcenia

Wiedza

- Student zna i rozumie pojęcia w zakresie pojęcia wirtualizacji i konteneryzacji z wykorzystaniem sieci komputerowych, ich technologii oraz protokołów komunikacyjnych.
- Student zna i rozumie pojęcia występujące w procesie konteneryzacji na przykładzie narzędzia Docker. Zna i rozumie proces tworzenia, skonteneryzowanych bezpiecznych,

warstwowych aplikacji internetowych; zna i rozumie pojęcia związane z orchestracją i monitorowaniem takiej aplikacji

Umiejętności

- Student potrafi potrafi ocenić przydatność wirtualizacji i konteneryzacji i związanych z nimi środowisk
- Student potrafi potrafi wyspecyfikować, zaprojektować, zaimplementować, przetestować oraz zdebuggować aplikację utworzoną w procesie konteneryzacji za pomocą plików Dockerfile.
- Student potrafi wytworzyć warstwową aplikację webową w oparciu o wybrane wzorce architektoniczne i przy pomocy narzędzia Docker utworzyć kontener i umieścić go w technologii chmurowej.
- Student potrafi zaplanować i przeprowadzić automatyczny proces tworzenia aplikacji z wykorzystaniem pojęcia konteneryzacji i związanych z nimi narzędzi.

Kompetencje społeczne

- Student jest gotów do samodzielnego uczenia się przez całe życie

7. Kryteria oceny

- wykład z elementami dyskusji z prezentacją multimedialną
- burza mózgów
- rozwijanywanie zadań
- analiza przypadków
- projekt praktyczny
- Kryteria oceny
- 100% Kolokwium pisemne zawierające do 10 zadań.
- 40% Ocena z laboratorium
- 60% Egzamin

8. Metody dydaktyczne

Wykład, laboratoria, praca własna studenta.

9. Literatura

Podstawowa:

- S. Kane, Docker: Up & Running, O'Reilly, 2024.
- J. Nickloff, S. Kuenzli, Docker in Action, Manning Publications, 2019.
- G. Schenker, Learn Docker – Fundamentals of Docker 19.x, Packt Publishing, 2020.

- E. Stoneman, Learn Docker in a Month of Lunches, Manning Publications, 2020.

Uzupełniająca:

- A. Davis, Bootstrapping Microservices with Docker, Kubernetes and Terraform, Manning Publications, 2021.
- E. Fouda, A Complete Guide to Docker for Operations and Development, Apress, 2022.
- F. Zammetti, Modern Full Stack Development using Typescript, React, Node.js, Webpack and Docker, Apress, 2020.

