



# PMSOFT

**Tytuł szkolenia:**

**Architektura aplikacji oparta o wirtualizacji Docker.**

**Liczba dni: 4 dni**

**Kod szkolenia:**

**architecture\_docker\_coreos.**

**Adresaci szkolenia:**

**Kierownicy projektów, devops, architekci, programiści..**

**Cel:**

**Zdobycie zdolności implementacji i korzystania z aplikacji poprzez kontenery dockerowe.**

## Motywacja szkolenia:

Lekka wirtualizacja dostarczona przez kernel linuxa poprzez cgroups umożliwia zbudowanie aplikacji jako zestaw niezależnych maszyn wirtualnych - tzw. kontenerów. Zarządzanie kontenerami jest mocno uproszczone poprzez możliwość wersjonowania oraz brak potrzeby dezinstalacji. Docker to dojrzałe narzędzie do zarządzania kontenerami, wspierające cały cykl życia kontenerów.

Pełne wykorzystanie możliwości kontenerów wymaga uruchomienie ich w infrastrukturze wspierającej klastry obliczeniowe. Kluczowa jest również pełna integracja z chmurami obliczeniowymi niezależnie od dostawców. Wsparcie tego typu jest dostarczone przez linuxową dystrybucją coreos.

## Zakres merytoryczny:

- Tworzenie obrazów dockera.
- Zarządzanie daemona dockera.
- Uruchomienie i backupowanie docker registry.
- Uruchomienie i zarządzanie instancjami kontenerów.
- Tworzenie i zarządzanie klastrem coreos.
- Orkiestracja kontenerów dockera na klastrze coreos za pomocą fleetctl.
- Podstawy dotyczące DNS-SD (DNS Service Discovery).

## Zakres technologiczny:

**Linux cgroups:** <https://www.kernel.org/doc/Documentation/cgroup-v1/>

**Docker:** <https://www.docker.com/>

**CoreOS:** <https://coreos.com/>

**Consul:** <https://www.consul.io/>

## Specyfikacja szkolenia:

Wymagania: Nie jest wymagana żadna wiedza wstępna.

Rodzaj szkolenia: 50% wykładów, 50% warsztaty.

Język materiałów: Angielski.

Język szkolenia: Angielski lub Polski.

## Program szkolenia:

### 1. Przegląd technologii wirtualizacji.

- Wirtualizacja przez Hypervisor.
- Wirtualizacja przez cgroup.
- Przegląd technologii cgroups w kernelu linuxa.

### 2. Docker

- Tworzenie Dockerfile.
- Budowanie obrazów dockera w maven i sbt.
- DockerHub i private docker registry.
- Zarządzanie demonami dockera.

### 3. Instancje dockera

- Zarządzanie lokalnymi obrazami dockera.
- Cykl życia instancji dockera.
- Interakcje z uruchomionymi instancjami dockera.

### 4. CoreOS

- Architektura CoreOS.
- Cloud-config.
- Tworzenie wirtualnego klastra coreos na Vagrant.
- Etcd2
- Flannel
- Fleet

### 5. Wdrożenie usług na CoreOS

- Tworzenie usług w oparciu o obrazy dockera.
- Orkiestracja przez fleetctl.
- Service Discovery przez DNS (consulDNS).
- Aktualizacje wersji aplikacji.
- Monitorowanie stanu klastra.

### 6. Wdrożenie w chmurze

- Przegląd dostawców.
- Wdrożenie w praktyce.