

PROJEKT UAIM

E-Recepta

Część 2

Marcin Latawiec, Pavel Klimuk, Jan Ignasiak,
Konrad Pióro

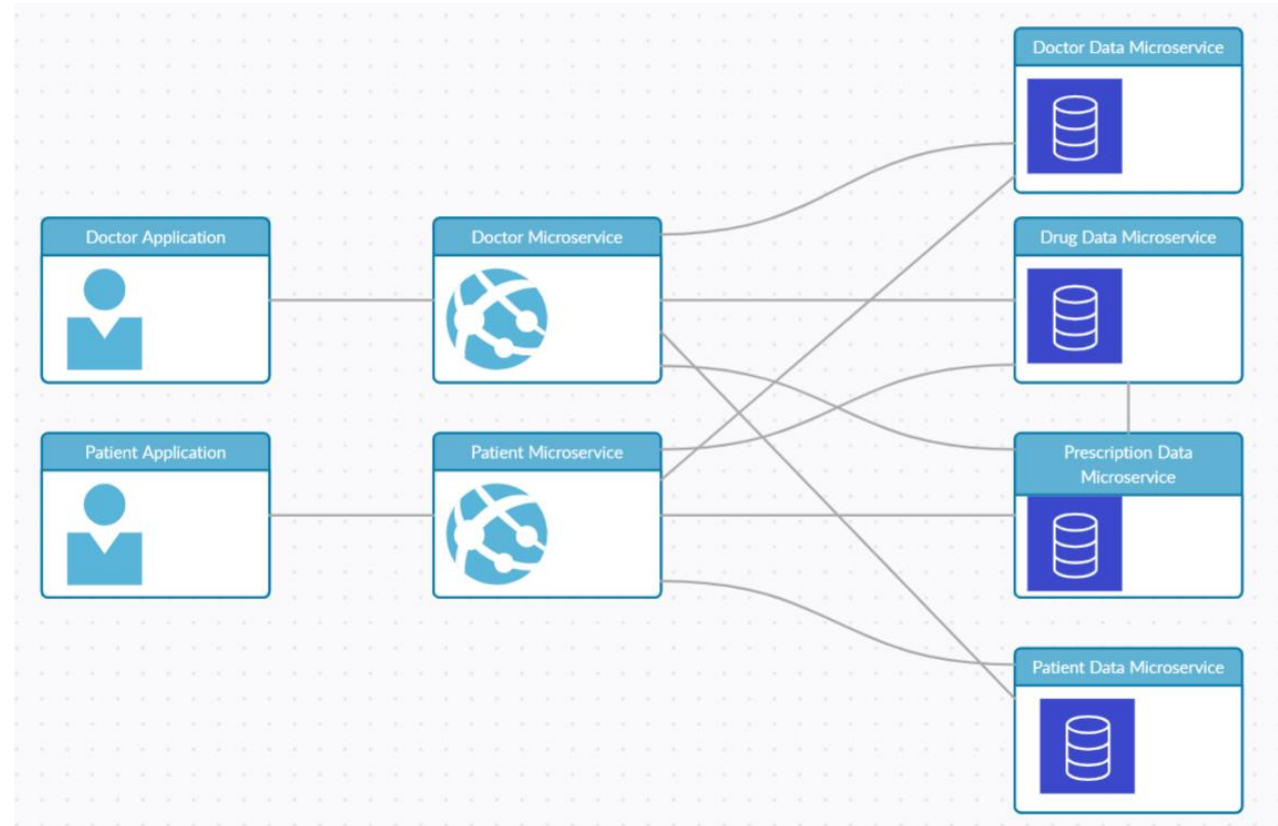


Projekty aplikacji użytkownika



Lista modułów

- Mikro-usługa danych recept
- Mikro-usługa danych leków
- Mikro-usługa danych pacjentów
- Mikro-usługa danych lekarzy
- Mikro-usługa aplikacji lekarza
- Mikro-usługa aplikacji pacjenta
- ☐ Aplikacja lekarza
- ☐ Aplikacja pacjenta



Specyfikacja mikroservisów

Dokumentacja specyfikacji poszczególnych modułów znajduje się w części 1 sprawozdania projektu.



CI/CD 1

- Zbudowano obrazy odpowiadające za następujące moduły systemu:
 - doctors-data-microservice – obraz mikrouслуги danych lekarzy
 - patients-data-microservice – obraz mikrouслуги danych pacjentów
 - drugs-data-microservice – obraz mikrouслуги danych leków
 - prescriptions-data-microservice – obraz mikrouслуги danych recept
 - doctors-application-microservice – obraz mikrouслуги aplikacyjnej lekarzy
 - patients-application-microservice – obraz mikrouслуги aplikacyjnej pacjentów
 - patients-client-application-microservice – obraz mikrouслуги klienckiej pacjenta
 - doctors-client-application-microservice – obraz mikrouслуги klienckiej doktora
- Zbudowane obrazy uruchamiane są przy użyciu docker-compose.
- W pliku docker-compose.yml znajdują się następujące dane:
 - Definicja sieci, z której korzysta platforma
 - Definicje serwisów – mikrouslug, które zobowiązujemy się uruchamiać
 - Przypisanie serwisów do powyższej sieci, nadanie im odpowiednich portów

CI/CD 2

1. Definicja przykładowego serwisu zdefiniowanego w pliku docker-compose.yml

```
doctors-application-microservice:
  container_name: "doctors-application-microservice"
  image: "uaimpw/doctor-application-microservice"
  ports:
    - "5204:80"
  environment:
    AppConfig__PatientDataBaseUrl: "http://patients-data-microservice"
    AppConfig__DoctorDataBaseUrl: "http://doctors-data-microservice"
    AppConfig__DrugDataBaseUrl: "http://drugs-data-microservice"
    AppConfig__PrescriptionDataBaseUrl: "http://prescriptions-data-microservice"
    ASPNETCORE_URLS: "http://+"
  networks:
    - "internal-network"
  restart: on-failure
  depends_on:
    - "doctors-data-microservice"
    - "patients-data-microservice"
    - "drugs-data-microservice"
    - "prescriptions-data-microservice"
  deploy:
    replicas: 1
```

2. W celu wdrożenia zbudowanych obrazów na platformie Kubernetes stworzone zostały pliki .yaml odpowiadające za wdrożenie poszczególnych mikrouslug. Przykładowy plik definiujący wdrożenie jednej z mikrouslug na platformie kubernetes

```
---
apiVersion: v1
kind: Service
metadata:
  name: doctors-data-service
spec:
  selector:
    app: doctors-data
  type: NodePort
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
      nodePort: 30000
```

3. W łańcuchu CI/CD zdefiniowanym w pliku *.gitlabci.yml* wyróżniliśmy odpowiednie sekcje odpowiadające za następujące akcje:

- Build – zbudowanie aplikacji
- Test – przeprowadzenie testów
- Docker – budowanie Obrazów
- DockerCompose – uruchamianie aplikacji na platformie docker-compose
- Kubernetes – uruchamianie aplikacji na platformie Kubernetes

Fazy procesu DevOps

1. Build

- Wywołanie odpowiednich skryptów budujących aplikację na platformie dotnet m.in. *dotnet build*

2. Test

- Uruchomienie mikroservisów na odpowiednich portach
- Przeprowadzenie testów
- „zabicie” mikroservisów

3. Docker

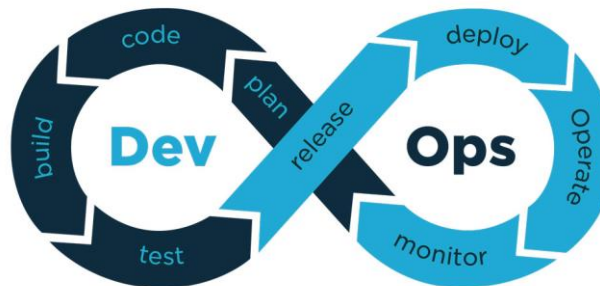
- Uruchomienie skryptów budujących obrazy m.in. *docker build*

4. Docker Compose

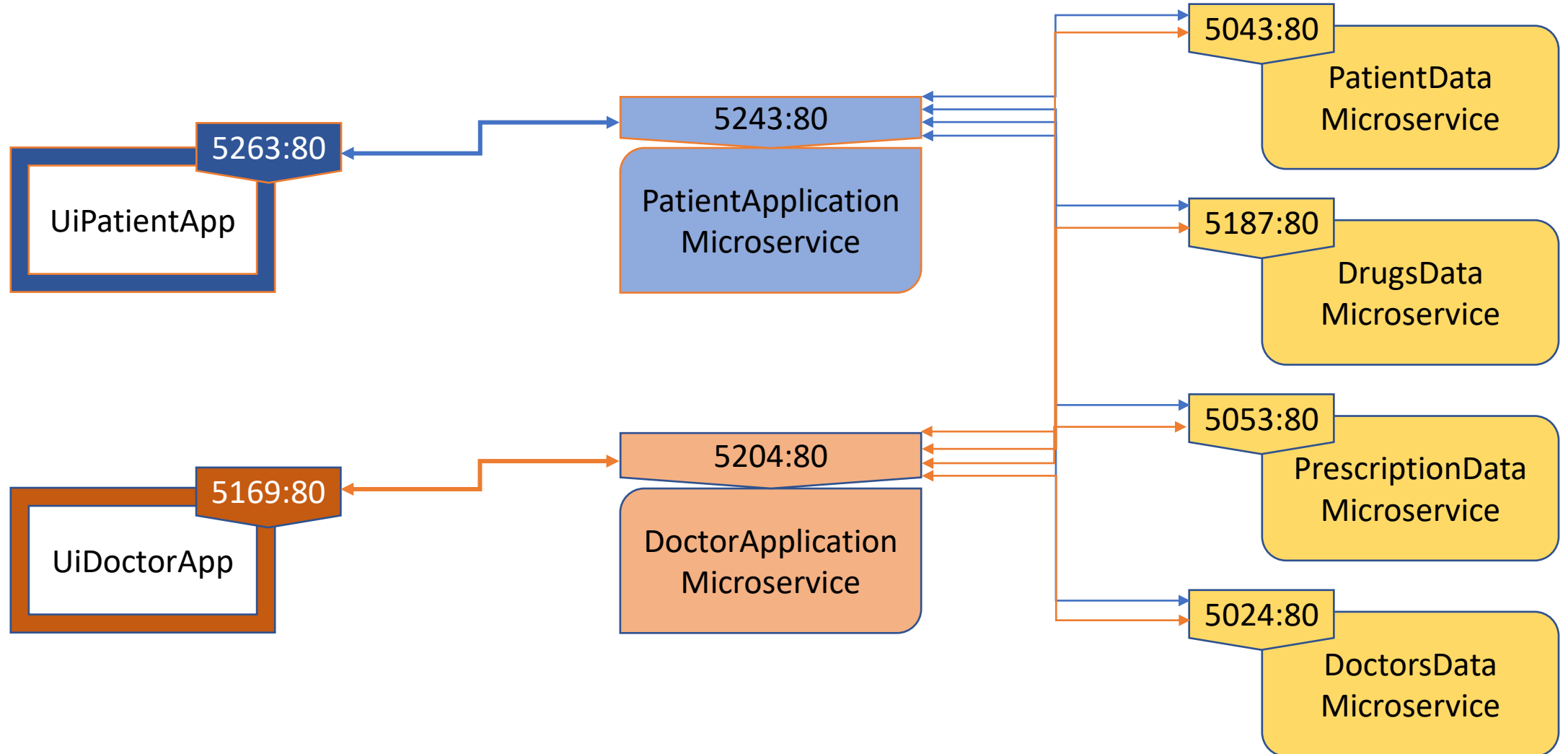
- Wywołanie skryptów uruchamiających system na platformie **docker-compose**, główna komenda: *docker-compose up -d*

5. Kubernetes

- wykonanie autorskich skryptów, mających za zadanie uruchamianie systemu na platformie **kubernetes**, wyświetlenie informacji o klastrze, a na końcu usunięcie serwisów – komenda *kubectl delete* :
 - k8s_startSystem.bat
 - k8s_showCluster.bat
 - k8s_stopSystem.bat



Schemat całej konfiguracji wdrożeniowej systemu.



Efekty końcowe 1

UiPatientApp

localhost:30006/listdrugs

About

Home

Counter

List Drugs

List Prescriptions

List Doctors

Settings

Help

Show Drugs:

Search

Drugs

Drug	Type	Needs Prescription
Xanax	CNS_Stimulant	True
Rutinoscorbin	Dietary_Supplement	False
Bioracef	Antibiotic	True

UiDoctorApp

localhost:30007/listpatients

About

Home

Counter

List Drugs

List Prescriptions

List Patients

Settings

Help

Show Patients:

Search

Patients

Patient FirstName	Patient LastName	Patient BirthDate	Patient Email	Pesel
Philipp	Star	06/06/2000	phstar@gmail.com	87071248476
Paul	Doe	08/01/1985	pdoe@gmail.com	48101179872
Jan	Kowalski	07/06/1993	jkow@gmail.com	50080945157

Efekty końcowe 2

The left screenshot displays the 'Images on disk' section of the Kubernetes Dashboard. It shows a table of 23 images with columns for NAME, TAG, IMAGE ID, CREATED, and SIZE. The table lists various images including aspneta, docker/desktop-storag..., docker/desktop-vpnkit..., docker/getting-started, hubproxy.docker.inter..., k8s.gcr.io/coredns/cor..., k8s.gcr.io/etcd, k8s.gcr.io/kube-apiser..., k8s.gcr.io/kube-contro..., k8s.gcr.io/kube-proxy, k8s.gcr.io/kube-sched..., k8s.gcr.io/pause, kubernetesui/dashboa..., kubernetesui/metrics..., uaiampw/doctor-applic..., uaiampw/doctor-client..., uaiampw/doctors-data..., uaiampw/drugs-data-mi..., uaiampw/patient-applic..., uaiampw/patient-dient..., uaiampw/patient data..., uaiampw/prescription d..., and ulapp.

The right screenshot displays the 'Workloads' section of the Kubernetes Dashboard. It shows a 'Workload Status' overview with three green circles representing Deployments, Pods, and Replica Sets. Below this, there are two tables: 'Deployments' and 'Pods'. The 'Deployments' table lists various deployments including doctor-client-application-deployment, doctors-application-deployment, patient-client-application-deployment, patients-application-deployment, prescriptions-data-deployment, doctors-data-deployment, drugs-data-deployment, and patients-data-deployment. The 'Pods' table lists various pods including doctor-client-application-deployment-5f9b4b6f8-7xflz, doctor-client-application-deployment-5f9b4b6f8-wm77, doctors-application-deployment-89779b29-paf58, doctors-application-deployment-89779b29-tatnh, prescriptions-data-deployment-fdcfb574-4dph, patients-application-deployment-79497c55f-12t6b, and patients-application-deployment-79497c55f-4d2b.