

Załącznik 1

Spis treści

1. Opis architektury	1
1.1.1. Umbraco CMS	1
1.1.2. Portal Mapowy – Backend.....	1
1.1.3. Portal Mapowy - pitmap-frontend.....	1
1.1.3. GeoServer	1
1.1.4. TaskManager - moduł kolejek	1
1.1.5. Geocode.....	2
1.1.6. Geocode Import	2
1.1.7. PIT Import.....	2
1.1.8. PIT Walidacja.....	2
1.1.9. PIT Notification.....	2
1.1.10. Kolejka RabbitMQ	2
1.1.11. GeoServer ConfigCheck.....	2
1.1.12. Usługa Uldk.....	2
1.1.13. Uke Authentication Service	2
2. Infrastruktura i wymagania systemowe	3
2.1. Serwery aplikacyjne	3
2.2. Serwery GIS	3
2.3. HAProxy dla Serwery bazodanowe	3
2.4. Serwery bazodanowe	3
2.5. Serwery CBO.....	3
2.6. Serwery Umbraco.....	3
2.7. Serwery Elasticsearch	3
3. Zarządzanie danymi.....	3
4. Skrypty i narzędzia.....	3
4.1. Portainer	3
4.2. Traefik.....	4
5. Monitorowanie	4
6. Inne	4

1. Opis architektury

1.1.1. Umbraco CMS

Technologia: .NET 6 MVC

Umbraco CMS – system CMS umożliwiający szybkie i łatwe tworzenie witryn internetowych i ułatwiający tworzenie złożonych aplikacji sieci web. Program Umbraco zawiera wielokrotnie nagradzane funkcje integracji i obsługuje w sposób macierzysty formaty niestandardowe i formanty użytkownika programu ASP.NET. System Umbraco to oprogramowanie na licencji Open-Source, wykorzystujące z silnik bazy danych MS SQL w wersji Express.

1.1.2. Portal Mapowy – Backend

Technologia: .NET 6 RestApi

Część backend systemu PIT Map została napisana we frameworku .NET 6, wykorzystuje bibliotekę Swagger, która publikuje dokładny opis punktów końcowych oraz kontrakty potrzebne do komunikacji pomiędzy częścią frontend i backend.

System integruje się z następującymi komponentami:

- moduł autentykacji (pit-modulaa)
- moduł autoryzacji (pit-modulaa - CheckUserPermission)
- serwisem kolejek (taskmanager)
- serwisem wystawiającym informacje pochodzące z modułu GeoServer (geocode-main)
- bezpośrednio z API usługi GeoServer
- modułem ULDK - system zwracający informację na temat wybranej działki
- serwerem SMTP do wysyłania poczty elektronicznej
- systemem Centralnej Bazy Organizacji

1.1.3. Portal Mapowy - pitmap-frontend

Technologia: VUE.JS

1.1.3.1. *pit-map*

Główne źródło danych dla aplikacji jest GeoServer.

1.1.3.2. *pit-admin-panel*

Technologia: .NET 7 + Blazor

1.1.3. GeoServer

Technologia: Java 11 / 17, Tomcat Server

GeoServer to oprogramowanie open source, które służy do udostępniania, zarządzania i przetwarzania danych geoprzestrzennych w formie usług internetowych, co umożliwia ich efektywne wykorzystanie w różnych aplikacjach związanych z przestrzenią geograficzną.

1.1.4. TaskManager - moduł kolejek

Technologia: .NET 6

Serwis odpowiedzialny za koordynację, w tym przyjmowanie oraz publikowanie wiadomości na kolejkę RabbitMQ. Przechowywanie oraz procesowanie wiadomości odbywa się przy użyciu klastra bazy danych.

1.1.5. Geocode

Technologia: .NET 6 RestApi

Serwis odpowiedzialny jest za pobieranie, zapisywanie i przetwarzanie danych wykorzystywanych do geokodowania oraz geokodowania odwrotnego.

1.1.6. Geocode Import

Technologia: .NET 6 RestApi

Usługa importuje dane adresowe udostępniane przez Główny Urząd Geodezji i Kartografii, a następnie zapisuje na serwerze bazodanowym.

1.1.7. PIT Import

Technologia: .NET 6 RestApi

Serwis jest odpowiedzialny przyjęcie danych użytkownika w postaci plików i umieszczenie ich na dedykowanym zasobie do dalszego przetwarzania np. Walidacji (w module PIT Walidacja) i importu danych do bazy.

1.1.8. PIT Walidacja

Technologia: .NET 6 RestApi

Serwis jest odpowiedzialny za walidowanie danych wejściowych / pliku wejściowego za pomocą reguł walidacyjnych, które są pobierane z bazy danych GIS.

1.1.9. PIT Notification

1.1.10. Kolejka RabbitMQ

Kolejka RabbitMQ we współpracy z usługą TaskManager została wykorzystana w module kolejek, wykorzystywanych w systemie do komunikacji między komponentami we współpracy z usługą TaskManager - moduł kolejek.

1.1.11. GeoServer ConfigCheck

Technologia: .NET 6

Przeznaczenie: Serwis odpowiedzialny jest za monitorowanie konfiguracji serwera GeoServer i w przypadku wykrycia zmian w konfiguracji, załadowanie nowych nastaw oraz zrestartowanie serwera GeoServer.

1.1.12. Usługa Uldk

Technologia: .NET 6 RestApi

Przeznaczenie: Serwis odpowiedzialny jest za zwracanie informacji o działce o zadanych parametrach (jako argumenty wyszukiwania) w formacie GeoJSON.

1.1.13. Uke Authentication Service

Technologia: .NET 6 RestApi

Przeznaczenie: moduły usługi Uke.Auth.Service czyli AAService oraz AuthService realizują zadania związane z pobieraniem informacji o użytkownikach, organizacjach oraz ich atrybutach jak również służą do wspierania mechanizmów logowania i wylogowywania oraz zarządzania tokenami dostępu.

2. Infrastruktura i wymagania systemowe

2.1. Serwery aplikacyjne

Serwery aplikacyjne skonfigurowane są w klaster: docker swarm i stanowią główną platformę uruchomieniową dla systemu PIT. Dodatkowo na hostach zainstalowany jest klaster rabbitmq

Serwery aplikacyjne połączone w klaster dla zwiększania HA są oparte na systemie Debian

2.2. Serwery GIS

Serwery z zainstalowanym narzędziem Geoserver.

Dla narzędzia Geoserver baza jest system operacyjny Debian z bazą Postgresql

2.3. HAProxy dla Serwery bazodanowe

Serwery stanowią proxy do baz danych. Skonfigurowane zostały 2 instancje w celu osiągnięcia wysokiej dostępności. Dostęp do serwerów jest realizowany przez VIP.

Na serwerach oprócz HAproxy zainstalowana jest baza typu key-value etcd który jest częścią konfiguracji wysoko dostępnego klastra Postgresql.

System operacyjny – Debian

2.4. Serwery bazodanowe

Serwery stanowią klaster bazodanowy dla silnika Postgresql. Skonfigurowane z wykorzystaniem narzędzia Patroni i bazy etcd

System operacyjny – Debian

2.5. Serwery CBO

Dla Centralnej Bazy Obiektów jest wykorzystany Windows Serwer

2.6. Serwery Umbraco

Dla serwera Umbraco jest wykorzystany Windows Serwer

2.7. Serwery Elasticsearch

Na serwerach elasticsearch zainstalowany jest klaster elasticsearch i kibana. Nie jest znane przeznaczenie tego klastra

System operacyjny – Debian

3. Zarządzanie danymi

Kontenery działają w klastrze docker swarm do katalogów na serwerze NFS. Udział serwera NFS jest podmontowany do wszystkich węzłów klastra.

Ponadto persystencja danych jest realizowana przez wolumeny, config i secrety dockerowe.

4. Skrypty i narzędzia

4.1. Portainer

Portainer to narzędzie do zarządzania kontenerami, które dostarcza intuicyjny interfejs graficzny do monitorowania, wdrażania i konfigurowania usług w klastrze Docker Swarm. Umożliwia łatwe zarządzanie wieloma węzłami oraz śledzenie stanu i wydajności kontenerów w zdecentralizowanym środowisku kontenerowym.

4.2. Traefik

Traefik to popularny narzędzie do routingu i zarządzania ruchem w środowiskach kontenerowych. W kontekście Docker Swarm, Traefik pełni rolę reverse proxy, umożliwiając obsługę wielu serwisów kontenerowych oraz dynamiczne zarządzanie nimi.

5. Monitorowanie

Monitoring środowisk został zrealizowany w oparciu o następujące komponenty:

- ✓ **Prometheus (v2.44)**: System monitorowania, zbiera metryki HTTP w modelu pull, przechowuje dane czasowe, umożliwia dynamiczne zapytania i alerty.
- ✓ **Grafana (v8.3.4)**: Platforma do wizualizacji danych, integruje się z różnymi źródłami, umożliwia konfigurację interaktywnych dashboardów.
- ✓ **cAdvisor (latest)**: Agent monitorujący dla kontenerów Docker, zbiera metryki użycia zasobów.
- ✓ **Node-exporter (latest)**: Eksporter metryk dla systemów Linux/Windows, dostarcza informacje o zasobach na poziomie hosta.
- ✓ **Elasticsearch-exporter (v1.5.0)**: Eksporter metryk dla Elasticsearch, oferuje informacje o wydajności indeksów i klastra.
- ✓ **Nginx-exporter (v0.10.0)**: Eksporter metryk dla serwera Nginx, dostarcza informacje o wydajności i obciążeniu.
- ✓ **Postgres-exporter (latest)**: Eksporter metryk dla PostgreSQL, dostarcza informacje o wydajności baz danych.

Ponadto aplikacje z wbudowanymi mechanizmami wystawiającymi metryki w formacie prometheus'a:

- ✓ **rabbitmq-server**: Metryki aplikacyjne dla rabbitmq
- ✓ **traefik**: Metryki aplikacyjne dla traefik
- ✓ **pitmap-be**: Metryki aplikacyjne dla aplikacji PIT 2.0 pitmap backend
- ✓ **import-metrics**: Metryki aplikacyjne dla aplikacji PIT 2.0 import

6. Inne

Tożsamość użytkowników dla systemów UKE w tym dla systemu PIT weryfikuje Centralny System Uwierzytelniania. CSU współpracuje z Węzłem Krajowy jako dostawca tożsamości.

Dane plikowe są przechowywane na NFSie montowanym do modułów wymagających dostępu do tego zasobu

Wszystkie ww. moduły pracują w środowisku zwirtualizowanym zarządzanym przez Biuro Informatyki UKE