Uniwersytet Gdański
Wydział Matematyki, Fizyki i Informatyki
Instytut Informatyki

Aplikacja do recenzji filmów

Konrad Kreczko

Projekt z przedmiotu technologie chmurowe na kierunku informatyka profil praktyczny na Uniwersytecie Gdańskim.

Gdańsk

26 czerwca 2024

Spis treści

1 Opis projektu

Aplikacja projektowa jest przeznaczona do zarzadzania ocenami filmów przez użytkowników. W ramach projektu zaimplementowano system bazujacy na architekturze mikroserwisów, w szczególności wykorzystujacy kontenery Docker oraz platforme Kubernetes do zarzadzania klastrami kontenerów.

1.1 Opis architektury

Architektura aplikacji oparta jest na mikroserwisach w kontekście Kubernetes. Główne komponenty architektoniczne to:

- **Backend**: Serwis obsługujacy logike biznesowa aplikacji, zaimplementowany jako kontener Dockerowy uruchamiany w klastrze Kubernetes.
- **Frontend**: Aplikacja kliencka napisana w React, komunikujaca sie z backendem poprzez API HTTP.
- **Baza danych**: PostgreSQL wykorzystywany do przechowywania danych o filmach, użytkownikach i ocenach.
- **Keycloak**: Serwer do zarzadzania tożsamościami i dostepem, zapewniajacy autoryzacje i uwierzytelnianie użytkowników.

Każdy komponent aplikacji jest wdrażany jako osobny Deployment w Kubernetes, co zapewnia skalowalność i niezawodność systemu.

1.2 Opis infrastruktury

Aplikacja działa lokalnie przy użyciu narzedzia Minikube do lokalnego zarzadzania klastrami Kubernetes. Minikube pozwala na uruchamianie jednoklastrowych środowisk Kubernetes na maszynie lokalnej.

Do uzyskania dostepu do aplikacji korzystano z port forwardingu, który przekierowuje ruch HTTP do odpowiednich usług w klastrze Kubernetes.

1.3 Opis komponentów aplikacji

Komponenty aplikacji sa dokładnie konfigurowane i zarzadzane przy użyciu Kubernetes:

- **Backend Deployment**: Kontener Dockerowy uruchamiany jako Deployment w Kubernetes, skalowany dynamicznie dzieki HorizontalPodAutoscaler.
- **Frontend Deployment**: Aplikacja React, umożliwiajaca interakcje użytkowników z backendem poprzez API REST.
- **PostgreSQL Deployment**: Baza danych PostgreSQL jako usługa używana do przechowywania danych aplikacji.
- **Keycloak Deployment**: Serwer Keycloak odpowiedzialny za zarzadzanie tożsamościami i autoryzacje użytkowników.

1.4 Konfiguracja i zarzadzanie

Konfiguracja aplikacji na poziomie klastra Kubernetes obejmuje ustawienia zasobów, takich jak CPU i pamieć, oraz zarzadzanie dostepem do danych aplikacji przez ConfigMaps i Secrets.

1.5 Skalowalność

Aplikacja jest skalowalna dzieki użyciu HorizontalPodAutoscaler, który monitoruje obciażenie CPU i automatycznie dostosowuje liczbe replik backendu w zależności od potrzeb.

1.6 Wymagania dotyczace zasobów

Dla każdego komponentu aplikacji określono wymagania dotyczace zasobów, takie jak minimalne i maksymalne użycie CPU oraz pamieci, zapewniajac odpowiednia wydajność i czas odpowiedzi dla użytkowników.

1.7 Architektura sieciowa

Architektura sieciowa aplikacji obejmuje konfiguracje w klastrze Kubernetes przy użyciu narzedzia Minikube do zarzadzania środowiskiem. Dostep do aplikacji uzyskiwany jest poprzez port forwarding, co umożliwia przekierowanie ruchu HTTP na odpowiednie porty usług w klastrze.

Literatura

[1] Docker - narzedzie służace do konteneryzacji komponentów aplikacji.

[2] Express.js - framework sieciowy użyty w cześci backendowej.

[3] Keycloak - system zarzadzania tożsamościa.

[4] keycloak-js - paczka użyta w cześci frontendowej i backendowej ułatwiajaca korzystanie z keycloak.

[5] Kubernetes - platforma służaca do orkiestracji aplikacji wielokontenerowych.

[6] Minikube - lokalny klaster kubernetesa użyty w projekcie.

[7] Postgresql - system bazodanowy użyty w projekcie.

[8] React - biblioteka użyta w cześci frontendowej aplikacji.