

Micro Train

Projekt polega na utworzeniu trzech mikroserwisów spełniających poniższe wymagania biznesowe.

Informacje wstępne

- Python 3.6 - 3.8
- Black (formatowanie)
- Flake8 (linter)
- Pytest (testy)
- Flask 1.0+
- SQLAlchemy
- Celery
- docker i docker-compose
- Dozwolone jest używanie zewnętrznych bibliotek

Opis zadania

Pociąg (mikroserwis 1)

Symuluje w sposób uproszczony pracę lokomotywy pociągu.

Mikroserwis rozgłasza w systemie kolejkowym:

- co 10s informację o aktualnej prędkości pociągu. Wartością jest liczba losowa z zakresu [0.0, 180.0]
- co 180s informację do której stacji zbliża się pociąg. Wartością jest losowa nazwa stacji ze zdefiniowanej listy. Proszę o zaszycie około 20 stacji. Przykładowe stacje:
<https://www.bazakolejowa.pl/index.php?dzial=stacje>

Centrala (mikroserwis 2)

Centralny ośrodek monitorujący pracę pociągów.

Na podstawie informacji rozgłaszanej przez pociąg realizuje poniższe reguły biznesowe:

Obsługa komunikatu o prędkości pociągu

- informacje o aktualnej prędkości pociągu wraz z aktualnym czasem zapisywane są do plików:
 - slow.log, gdy prędkość w zakresie [0.0, 40.)
 - normal.log, gdy prędkość w zakresie [40.0, 140.)

- fast.log, gdy prędkość w zakresie [140.0, 180.0]

Obsługa komunikatu o zbliżaniu się do stacji

- informacja o zbliżaniu się do stacji logowana jest przez moduł logging i powinna zawierać informację o stacji, do której się zbliża się pociąg i aktualny czas.
- w momencie otrzymania informacji o zbliżaniu się pociągu do stacji, mikroserwis odpytuje Dróżnika o stan szlabanu.
 - jeżeli jest otwarty, wysyła informację do dróżnika, aby opuścić szlaban
 - jeżeli jest zamknięty, loguje anomalię i przechodzi do kolejnego punktu
- po 10s od zamknięcia szlabanu, wysyła informację do dróżnika, aby podnieść szlaban
- informacje o podniesieniu i opuszczeniu szlabanu logowane jest z poziomem INFO

Komunikacja z Dróżnikiem odbywa się poprzez REST+JSON.

Dróżnik (mikroserwis 3)

Symuluje w sposób uproszczony pracę dróżnika pilnującego przejazdów kolejowych.

Mikroserwis udostępnia interfejs REST+JSON umożliwiający:

- sprawdzenie aktualnego stanu szlabanu (otwarty/zamknięty, informacja o ostatniej zmianie)
- zmianę aktualnego stanu szlabanu

Mikroserwis przechowuje w bazie danych informację o stanie szlabanu oraz informację o ostatniej zmianie.

Na co zwracać uwagę:

- kod powinien posiadać README.md z prostą instrukcją umożliwiającą uruchomienie aplikacji
- komunikacja REST powinna wykorzystywać prawidłowe czasowniki HTTP
- komunikacja REST powinna wykorzystywać prawidłowe statusy odpowiedzi
- aplikacja powinna posiadać podstawowe testy jednostkowe
- kod powinien być sformatowany z wykorzystaniem Black
- kod powinien być sprawdzony linterem flake8
- kod powinien zostać stworzony zgodnie z dobrymi praktykami, w szczególności dotyczącymi bezpieczeństwa, niezawodności i czytelności

Jak oddać zadanie

1. Gotowe repozytorium pakujemy komendą: `git bundle create IMIĘ_NAZWISKO.bundle --all`.
Proszę pamiętać aby wszelkie zmiany były za-commitowane (!)
2. Paczkę `.bundle` wysyłamy jako załącznik na mój email.

Plusem będzie zastosowanie Static Typing.

W razie pytań, proszę o kontakt.