



### 1. Opis zadania

Zadanie polega na stworzeniu mikroserwisu wspierającego pracę programistów zajmujących się uczeniem maszynowym. System ma pomóc w gromadzeniu i udostępnianiu informacji pobranych z sieci. Główną funkcjonalnością systemu jest pobieranie tekstu oraz obrazków ze stron internetowych.

### 2. Funkcjonalność

- Zlecenie pobrania tekstu z danej strony internetowej i zapis jej w systemie.
- Zlecenie pobrania wszystkich obrazków z danej strony i zapis ich w systemie.
- Sprawdzenie statusu zleconego zadania.
- Możliwość pobrania stworzonych zasobów (tekstu i obrazków).

### 3. Architektura

- Zadanie polega na zaprojektowaniu i zaimplementowaniu REST API dla tego systemu. Mikroserwis powinien być napisany w języku Python.
- Rozwiązanie powinno zawierać testy automatyczne.
- Uruchomienie mikroserwisu powinno być maksymalnie zautomatyzowane (preferowane użycie Dockera lub podobnych narzędzi).

### 4. FAQ

- Czy wymagane jest wykonanie Javascriptu w celu uzyskania tekstu/obrazków na stronie? **Nie, pobieramy tylko statyczne zasoby.**
- Czy z tekstu pobieranego ze stron powinien usuwać tagi HTML i kod Javascript? **Tak.**
- Czy napisanie frontendu jest częścią zadania? **Nie.**
- Czy można założyć, że pobieranie tekstu/obrazków ze strony jest szybkie? **Nie, pobieranie może trwać bardzo długo.**

### 5. Kryteria sukcesu:

- Właściwa architektura dla tego problemu
- Poprawnie zaprojektowane i zaimplementowane API (niekoniecznie dogłębna implementacja)
- Automatyzacja systemu
- Testy systemu
- Kod dostarczony jako udostępnione repozytorium gita (np. w BitBucket)
- Pisemny komentarz od autora na temat rozwiązania. Może być w stylu retro (co poszło ok, co nie ok, co do zmiany)

**Odpowiednia architektura systemu i design API są ważniejsze niż dogłębna implementacja.**