

Twoje zadanie polega na opracowaniu w języku Python aplikacji WEB, która będzie realizować poniższe wymagania:

Wymagania funkcjonalne

- Użytkownik przez stronę internetową musi mieć możliwość podania listy punktów (lokalizacji geograficznych), po jej zatwierdzeniu zostanie wyświetlona sumaryczna odległość [metry] z podanej listy (ścieżki) oraz czas wykonywania obliczeń [sekundy]. Interfejs musi umożliwiać podanie do 50 punktów geograficznych w ścieżce.
- Aplikacja udostępni API przez które można się z innego programu odpytać o odległość ścieżki. API jako parametry przyjmie:
 - identyfikator zapytania REQUEST_ID (string)
 - listę punktów geograficznych, współrzędne podane w stopniach (float), a jako wynik zwróci:
 - sumaryczną odległość z podanej listy punktów [metry]
 - czas wykonywania obliczeń [s].
 - API musi umożliwiać przyjęcie do 50 punktów geograficznych.
- Wykonywanie przez API obliczenia należy zapisywać w celach statystycznych. Należy zapamiętać:
 - podany w zapytaniu identyfikator zapytania REQUEST_ID,
 - czas (timestamp) rozpoczęcia obliczeń,
 - czas (timestamp) zakończenia obliczeń.

Nie możemy przechowywać zduplikowanych identyfikatorów zapytań REQUEST_ID, ale musimy umożliwić wielokrotnie wykonywanie zapytań z tym samym REQUEST_ID. W takim wypadku nadpisujemy czasy rozpoczęcia i zakończenia.

Wymagania niefunkcjonalne

- Wynik obliczeń wykonywanych przez interfejs użytkownika oraz przez API zostanie zwrócony w czasie maksymalnie 10 sekund niezależnie od ilości podanych punktów.
- Opracowanie dokumentacji do API Twojego serwisu. Może być w postaci pliku Markdown. Tak aby można to było przekazać innej osobie aby widziała jak korzystać z Twojego serwisu.
- Do obliczania odległości pomiędzy dwoma kolejnymi punktami należy korzystać z udostępnionego serwisu GeoDistance.
- Sumaryczną odległość liczymy jako sumę odległości pomiędzy kolejnymi punktami.
$$\text{sum_dist} = \text{dist}(P1,P2) + \text{dist}(P2,P3) + \dots + \text{dist}(Pn-1, Pn)$$

Informacje o serwisie GeoDistance

- Serwis GeoDistance dla podanych dwóch punktów geograficznych zwraca odległość pomiędzy nimi w metrach.
- Serwis udostępnia API w postaci HTTP REST. Na podane w przykładzie zapytanie zwraca w wynik w formacie json.

Treść przykładowego zapytania do serwisu GeoDistance:

```
> curl -u "Cristoforo:Colombo"  
"http://146.59.46.40:60080/route?origin=50.0,18.1&destination=50.2,18.3"
```

Przykładowa odpowiedź z serwisu GeoDistance:

```
{"distance":34335.926162997166}
```

Oczekiwany rezultat

- Program, serwis który możemy uruchomić i zobaczyć jak działa od strony użytkowej.
- Dostęp do repozytorium GIT z historią commit-ów taką, żeby można było prześledzić proces dochodzenia do rozwiązania, a nie tylko wynik końcowy.

Jeżeli w trakcie pracy wystąpią jakieś problemy, wątpliwości do co wymagań, serwis GeoDistance będzie niedostępny to można się kontaktować.