Dokumentacja techniczna:

Plik delta w dalszym dokumencie rozumiany jest przez plik różnicowy w formacie .csv

Metody Public API:

|  |  |  |  |
| --- | --- | --- | --- |
| Nazwa metody | Wejście | wyjście | Opis |
| receiveUpdates | Plik delta | True/False | Dane są zmienne:  **sz** -- start zakresu  **kz** -- koniec zakresu  **sip** -- adres sip  **data** -- data zmiany  **for**  *sz*,*kz*,*sip*,*data* **in**  csv  *zak* = zakres(*sz*,*kz*)  WstawZakres\_Z(*zak*)  WstawZakres\_Z(*zak*){  **for** *zak'* **in** ZakresyPokrywajaceSie(*zak*): (SELECT)  **if** *zak* pokrywa sie z lewej strony z *zak'*:  wydziel *zak'*-*zak* = (*sz'*, *sz-1*) (UPDATE)  **elif** *zak* pokrywa sie z prawej strony z *zak'*:  Wydziel(*z'*-*z*) = (*kz+1*, *kz'*) (UPDATE)  **elif** *zak* pokrywa w calosci *zak'*:  usun *zak'* (DELETE)  WstawDoBazy() (INSERT)  *// Możliwa jest optymalizacja:*  *// Jeżeli w pokrywającym zakresie dane są identyczne jak w nowym*  *// to nic nie rób*  }  *// sa, ka – start, koniec zakresu*  *// a – zakres*  ZakresyPokrywajaceSie(zakres *zak*){  a = Zakres(*sa*,*ka*)  SELECT start,koniec  FROM zakresy  WHERE start BETWEEN *sa* AND *ka*  OR koniec BETWEEN *sa* AND *ka*  }  *// a – zakres do zmiany*  *// b – nowy zakres*  *// sa, ka, sb, kb – start, koniec odpowiednio zakresów a i b*  Wydziel(zakres zak){  UPDATE zakresy  SET start = *sb*, koniec =*kb*, data = NOW(), …  WHERE start = *sa*  }  WstawDoBazy(){  INSERT INTO zakresy  (start, koniec, data, sip, ...)  VALUES (sz, kz, data, sip, ...)  } |
| sendData | --- | Plik delta | SELECT start, koniec, sip, data  FROM zakresy |
| sendUpdates | data(data od której wysłać modyfikacje) | Plik delta | SELECT start, koniec, sip, data\_zmiany  FROM zakresy  WHERE data\_zmiany > data |

Metody Admin API:

|  |  |  |  |
| --- | --- | --- | --- |
| Nazwa metody | Wejście | Wyjście | Opis |
| modifyRange | Zakres ‘z’ | True/False | WstawZakres\_Z(*zak*);  *//patrz opis receiveUpdate* |
| deleteRange | Zakres ‘z’ | True/False | Usuń zakres (DELETE) |
| addRange | Zakres ‘z’ | True/False | WstawZakres\_Z(*zak*);  *//patrz opis receiveUpdate* |

­­

Korzystanie z aplikacji klienckiej: (numbex\_client.py)

|  |  |
| --- | --- |
| Komenda | Opis działania |
| pull <data> | Aktualizacja danych począwszy od daty <data> oraz wysłanie ich na webservice (sendUpdates(<data>)) |
| pull –o <nazwa\_pliku> | Zapis różnic (csv) do pliku <nazwa\_pliku>  Jeżelki <nazwa\_pliku> jest pusta to dane wyświetlane są na stdout |
| pullall | Ściągnięcie wszystkich danych z bazy i wysłanie ich na webservice (sendData()) |
| pullall – o <nazwa\_pliku> | Zapis różnic (csv) do pliku <nazwa\_pliku>  Jeżelki <nazwa\_pliku> jest pusta to dane wyświetlane są na stdout |
| push <plik\_csv> | Wysłanie wszystkich danych na serwer (receiveUpdates(<plik\_csv>) na webserwisie) |
| -u <URL> | Przy starcie; dodanie adresu <URL> webservice’u  Jeżeli <URL> jest pusty to ustawiany jest adres http://localhost:8000/ |

Korzystanie z aplikacji serwerowej: (numbex\_server.py)

|  |  |
| --- | --- |
| Komenda | Opis działania |
| -l | Przy starcie; loglevel; poziom logów; <DEBUG|WARN> |
| -p | Przy starcie; port; port na którym działa serwer; <PORT> |
| -d | Przy starcie; database; baza danych sqlite; <plik\_z\_baza\_danych>  Jeżeli plik nie istnieje, aplikacja stworzy nową bazę. |