

Przygotował: NNMF

Zatwierdził: OOCL

Autoryzował: ALMO

Kod: GMV-TRISTAR-TN-015

Kod wewnętrzny:

Wersja 1

Data: 14/10/2013

## STATUS DOKUMENTU

|  |   |
|--|---|
| 1. WPROWADZENIE .....                    | 4 |
| 1.2. DEFINICJE I SKRÓTY.....             | 4 |
| 1.2.1. DEFINICJE .....                   | 4 |
| 1.2.2. SKRÓTY .....                      | 4 |
| 2. ODNIESIENIA .....                     | 5 |
| 2.1. ISTOTNE DOKUMENTY .....             | 5 |
| 2.2. DOKUMENTY POMOCNICZE .....          | 5 |
| 3. WYMIANA DANYCH .....                  | 6 |
| 3.1. DANE TOPOLOGII .....                | 6 |
| 3.1.1. FUNKCJE USŁUGI SIECIOWEJ.....     | 6 |
| 3.1.2. PRZEPŁYW DANYCH.....              | 6 |
| 3.2. DANE DOTYCZĄCE ROZKŁADU JAZDY ..... | 6 |
| 3.2.1. FUNKCJE USŁUGI SIECIOWEJ.....     | 6 |
| 3.2.2. PRZEPŁYW DANYCH.....              | 8 |

## WYKAZ RYSUNKÓW I TABEL

Skróty 4

Tabela 2-1 Istotne dokumenty 5

5

## 1. WPROWADZENIE

Projekt Tristar

### 1.2. DEFINICJE I SKRÓTY

#### 1.2.1. DEFINICJE

Pojęcia i terminy użyte w niniejszym dokumencie, które uznano za właściwie zdefiniować, zamieszczono w poniższej tabeli:

**Tabela 2-1 Definicje**

#### 1.2.2. SKRÓTY

Skróty **Tabela 2-2 Skróty**

| Skrót | Definicja |
|-------|-----------|
|       |           |
|       |           |
|       |           |
|       |           |

## 2. ODNIESIENIA

### 2.1. ISTOTNE DOKUMENTY

Poniższe dokumenty, szczególnie we wskazanej wersji, tworzą część niniejszego dokumentu w takim stopniu w jakim są przez niego określone. Za dokumenty istotne, uważa się takie, do których odnosi się Umowa, lub są zatwierdzone przez Organ Zatwierdzający, jako dokumenty istotne. W tym dokumencie mają one format [DA.x]:

**Tabela 2-1 Istotne dokumenty**

| Ref.   | Tytuł | Kod | Wersja | Data |
|--------|-------|-----|--------|------|
| [DA.1] |       |     |        |      |

### 2.2. DOKUMENTY POMOCNICZE

Poniższe dokumenty, chociaż nie tworzą części niniejszego dokumentu, poszerzają lub wyjaśniają jego zawartość. Za dokumenty pomocnicze, uważa się wszystkie te dokumenty, które nie są istotne i do których odnosi się niniejszy dokument. Mają one format [DS.x]:

**Tabela 2-2 Dokumenty pomocnicze**

| Ref.   | Tytuł | Kod | Wersja | Data |
|--------|-------|-----|--------|------|
| [DS.1] |       |     |        |      |
|        |       |     |        |      |

## 3. WYMIANA DANYCH

### 3.1. DANE TOPOLOGII

Firma GMV udostępni web service, tak by można pobierać wersje topologii. Istnieją różne wersje topologii w zależności od danych, należy więc wskazać numer wersji podczas pobierania danych topologii.

#### 3.1.1. FUNKCJE USŁUGI SIECIOWEJ

##### **TopologyVersion[] GetTopologyVersion(int fleet)**

Wej.:

- fleet: int

Wyj.:

- TopologyVersion:
  - o VersionNumber: int
  - o StartDate: DateTime
  - o EndDate: DateTime

#### 3.1.2. PRZEPŁYW DANYCH

W pierwszej kolejności należy wywołać funkcję *GetTopologyVersion*, aby pobrać wersje topologii dostępne dla danej daty W najnowszej wersji topologii data końcowa będzie równa *null*.

## 3.2. DANE DOTYCZĄCE ROZKŁADU JAZDY

Firma GMV udostępni web service dla importowania rozkładów jazdy. Należy wywołać usługę sieciową, gdy zostanie utworzony nowy rozkład jazdy, lub aby przypisać autobusy do usługi.

#### 3.2.1. FUNKCJE USŁUGI SIECIOWEJ

##### **Report ImportNewTimetable(ImportHHPP timetable)**

Wej.:

- ImportHHPP
  - o Fleet: int
  - o TopologyVersionNumber: int
  - o TimeTableName: string(50) (nazwa rozkładu jazdy. Unikalny klucz)
  - o StartDate: DateTime (wymagana tylko data)
  - o EndDate: DateTime (wymagana tylko data)
  - o DaysOfWeek: int
    - Jeśli aktywne w poniedziałki: 1, else 0; we wtorki: 2, else 0; w środy: 4; w czwartki: 8; w piątki: 16; w soboty: 32; w niedziele: 64. Końcowa wartość stanowi sumę tych wszystkich wartości.
  - o Expedition[]: Tablica obiektów Expedition (kursów)
  - o Overwrite: Funkcja logiczna (jeśli prawdziwa, nadpisuje istniejące rozkłady jazdy dla tej daty)
- Expedition
  - o LinelId: int
  - o RoutelId: int
  - o BusServiceName: string(8)

- DriverServiceName: string(8)
- Order: int (kolejność kursu w zadaniu pojazdu(brygadzie))
- BusStopTime[]: Tablica obiektów BusStopTime
- BusStopTime
  - BusStopId: int (Id przystanku)
  - Order: int (kolejność przystanku w danym kursie)
  - PassingTime: DateTime (wymagany tylko czas – czas przejazdu przez przystanek)

Wyj.:

- Report
  - Result: Funkcja logiczna (prawdziwa, jeśli dane są prawidłowe, fałszywa, jeśli dane są nieprawidłowe)
  - ErrorList[]: tablica stringów (w przypadku błędów, funkcja ta stanowi wytłumaczenie błędów)

**Report ImportAssignations(int fleet, string timeTableName, DateTime startDate, DateTime endDate, int DaysOfWeek, Assignment[] busAssignmentData, Assignment[] driverAssignmentData, bool overwrite)**

Wej.:

- fleet: int
- timeTableName: string(50) (nazwa rozkładu jazdy utworzonego wcześniej)
- startDate: DateTime (wymagana tylko data)
- endDate: DateTime (wymagana tylko data)
- DaysOfWeek: int – ta sama zależność jak w funkcji ImportNewTimetable.
- busAssignmentData[]: Tablica obiektów przypisać
- driverAssignmentData[]: Tablica obiektów przypisać
- overwrite: Funkcja logiczna (jeśli prawdziwa, nadpisuje istniejące przypisania dla tych dat)
- Assignment:
  - ServiceName: ciąg(8) (nazwa usługi autobusowej lub kierowcy utworzonych wcześniej)
  - ResourceName: ciąg(4) (kod autobusu lub kierowcy)

Wyj.:

- Report
  - Result: Funkcja logiczna (prawdziwa jeśli dane są prawidłowe, fałszywa, jeśli dane są nieprawidłowe)
  - ErrorList[]: tablica stringów (w przypadku błędów, funkcja ta stanowi wytłumaczenie błędów)

**ExportHHPP GetTimetable(DateTime date, int fleet, int TopologyVersionNumber)**

Wej.:

- Data: DateTime
- Fleet: int
- TopologyVersionNumber: int

Wyj.:

- ExportHHPP
  - TimeTableName: string(50) (nazwa rozkładu jazdy)
  - DaysOfWeek: int

- Jeśli aktywne w poniedziałki: 1, else 0; we wtorki: 2, else 0; w środy: 4; w czwartki: 8; w piątki: 16; w soboty: 32; w niedziele: 64. Końcowa wartość stanowi sumę tych wszystkich wartości.
- Expedition[]: Tablica obiektów trasy
- Expedition
  - LineId: int
  - RouteId: int
  - BusServiceName: ciąg(8)
  - DriverServiceName: ciąg(8)
  - Order: int (zamówienie trasy w usłudze autobusowej)
  - BusStopTime[]: Tablica obiektów BusStopTime
- BusStopTime
  - BusStopId: int
  - Order: int
  - PassingTime: DateTime (wymagany tylko czas)

### 3.2.2. PRZEPŁYW DANYCH

Każdorazowo, gdy następuje zmiana w rozkładzie jazdy informacja ta powinna być wysłana do systemu SAE poprzez wywołanie funkcji *ImportNewTimeTable*. Każdy rozkład posiada unikalną nazwę i powinien odnosić się od określonej wersji topologii. Istnieje możliwość, że niektóre rozkłady jazdy są aktywne dla tej samej wersji topologii, przykładowo, różne rozkłady jazdy na dni powszednie, soboty lub niedziele. Ważne jest, by każdorazowo funkcja ta była wywoływana; cały rozkład jazdy dla wszystkich linii musi być wysłany jednorazowo. Przykładowo, jeśli zdefiniowana jest linia w wersji topologii, ale brak jest danych dla tej linii w wysłanym rozkładzie jazdy, oznacza to, iż linia nie będzie aktywna dla tego rozkładu jazdy.

Ponadto w przypadku nowej wersji topologii, należy zaimportować do tej wersji nowy rozkład, w tym cały rozkład jazdy, nawet w przypadku linii, które nie są w ogóle modyfikowane.

Po udanym eksporcie rozkładu jazdy do systemu SAE, użytkownik może wysłać przypisania zadań transportowych (brygad) dla każdego dnia. Dokonuje się tego poprzez wywołanie funkcji *ImportAssignations*. Te przypisania zadań transportowych (brygad) należy wysłać dzień wcześniej, tak by serwer SAE mógł odczytać je przed rozpoczęciem danego dnia.

Obie funkcje zwrócą wartość wskazującą, czy import danych powiódł się bądź nie oraz wskazując opis błędów, jeśli import się nie powiódł.

W przypadku wystąpienia błędnych danych, dane prawidłowe zostaną zapisane w systemie SAE.

Aby pobrać aktywne rozkłady jazdy, należy w pierwszej kolejności wywołać funkcję *GetTopologyVersion*, aby pobrać wersje topologii dostępne dla każdej daty. W najnowszej wersji topologii data końcowa będzie równa *null*. Następnie należy wywołać funkcję *GetTimetable*, wskazując datę, flotę i wybraną wersję, by móc pobrać te dane.



KONIEC DOKUMENTU