

Dokumentacja interfejsu ITS

Interfejs komunikacyjny oparty o technologię RESTful Web Services dla systemu ITS we Wrocławiu pozwalający na zasilanie Repozytorium Danych ITS informacjami pochodzącymi z pojazdów Operatora

Wersja:	1.0
Ostatnio zmodyfikowano:	11-05-2021

1. Słownik pojęć

Skrót / pojęcie	Definicja
APN	Access Point Name (APN) - nazwa wskazująca na konkretną sieć pakietową (np. intranet klienta).
BUSMAN	Aplikacja do konstruowania/tworzenia rozkładów jazdy dla pojazdów transportu publicznego
BUSMAN CB	Baza relacyjna MSSQL, służąca do składowania opracowanych wcześniej rozkładów jazdy przy pomocy oprogramowania Busman
DIP	Dynamiczna Informacja Przystankowa.
Dostawca danych lub Dostawca danych lokalizacyjnych	Właściciel pojazdów (przewoźnik) wyposażonych w komputery pokładowe, który jest odpowiedzialny za realizację rozkładu jazdy
Entry	Numer logiczny detektora stosowany w logice powiadomień przekazywanych przez sterownik sygnalizacji świetlnej PLC do systemu sterowania ITS.
ITS	System Inteligentny System Transportu, którego właścicielem jest ZDiUM Wrocław
Lokalizacja GPS	Współrzędne geolokalizacyjne x,y w układzie odniesienia WGS84
MIT	Moduł Integratora Danych Transportowych – jeden z modułów systemu ITS
Pojazd transportu publicznego	Pojazd wyposażony w komputer pokładowy realizujący rozkład jazdy
Punkt detekcji	<i>Patrz:</i> Strefa Detekcji
Radio krótkiego zasięgu	Odbiornik radiowy skojarzony ze sterownikiem PLC na skrzyżowaniu, do którego wysyłane jest zgłoszenie przejazdu

Skrót / pojęcie	Definicja
Ramki asynchroniczne	Ramki zdarzeniowe przesyłane w ściśle określonej sytuacji / po wystąpieniu konkretnego zdarzenia. identyfikator zdarzenia większy od 0 (opisany w rozdziale „Opis zasobów zdarzeń”)
Ramki synchroniczne	Ramki lokalizacyjne przesyłane z pojazdów z minimalną częstotliwością co 15s (lub 2s w zasięgu punktów zgłoszeniowych wariantu C). identyfikator zdarzenia równy 0 (opisany w rozdziale „Opis zasobów zdarzeń”)
Rekord (ramka) danych pojazdu	Ramka synchroniczna lub asynchroniczna wysyłana przez komputer pokładowy pojazdu
Rozkład jazdy	Rozkład jazdy pojazdów transportu publicznego, przygotowanego przez WTR z wykorzystaniem oprogramowania Busman
Strefa detekcji	Lokalizacja geograficzna (punkt gps wraz z promieniem w metrach), w której zgodnie z definicją komputery pokładowe są zobowiązane do wysyłania zgłoszeń priorytetu
System ITS	System służący do sterowania i zarządzania ruchem w mieście Wrocław.
System Sterowania Ruchem	System dynamicznego sterowania ruchem kołowym zarządzający sygnalizacją świetlną. Jednostkami wykonawczymi są sterowniki sygnalizacji świetlnej PLC.
Trasa alternatywna	Trasa wyznaczona przez system ITS pozwalająca na ominięcie utrudnienia
Wariant A	Wariant zgłaszania priorytetu przejazdu dla linii tramwajowej. Dane wejściowe stanowią ramki lokalizacyjne przesyłane co 15 sekund oraz zgłoszenia priorytetu, które są powiązane z segmentem trasy (parą słupków) rozkładu jazdy
Wariant B	Wariant zgłaszania priorytetu przejazdu (autobus lub tramwaj) dla pojazdów posiadających komputery pokładowe zasilone wirtualnymi punktami zgłoszeniowymi niezależnymi od rozkładów jazdy. Pojazdy niezależnie od realizowanej trasy zgłaszają żądanie priorytetu w każdym obszarze zdefiniowanym w tym wariantcie
Wariant C	Wariant zgłaszania priorytetu przejazdu (autobus) linii przejazdu linii priorytetowej. Komputery pokładowe zmieniają częstość nadawania ramek synchronicznych do systemu ITS z 15s na 2s (jeśli znajdują się wewnątrz zdefiniowanych obszarów punktów zgłoszeniowych)
Wariant detekcji	Uzgodniony z Zamawiającym wariant zgłaszania wariantu w wersji A,B lub C dla pojedynczego pojazdu

Skrót / pojęcie	Definicja
Zgłoszenie, Zgłoszenie priorytetu	Żądanie przesyłane do systemu ITS w celu umożliwienia przejazdu linii priorytetowej wysyłane przez komputer pokładowy pojazdu

2. Wstęp

Dokumentacja opisuje interfejs komunikacyjny RESTful umożliwiający zasilanie danymi pochodzącymi z pojazdów transportu publicznego miejskiego przewoźnika oraz zewnętrznych przewoźników.

Interfejs ten został zaprojektowany w taki sposób, aby możliwa była komunikacja:

- komputery pokładowe – serwer centralny,
- serwer centralny przewoźnika – serwer centralny.

W dokumencie opisano również wszystkie warianty zgłaszania priorytetu do systemu ITS, które pozwalają na prawidłową obsługę pojazdów transportu publicznego przez system ITS.

2.1 Ogólne zasady komunikacji

Architektura rozwiązania mechanizmu funkcji API zrealizowana jest z wykorzystaniem technologii RESTful z zastosowaniem formatu JSON do opisu operacji oraz parametrów. Jako protokół transportowy wykorzystywany jest protokół transmisji HTTPS (transmisja szyfrowana). Dzięki zastosowaniu otwartego protokołu system ITS dostarcza interfejsy komunikacyjne do wykorzystania dla integracji pojazdów transportu publicznego.

Podstawowym wymaganiem jest przekazywanie bieżących informacji o lokalizacji pojazdu transportu publicznego z częstotliwością nie mniejszą niż 15 sekund oraz ramek zdarzeniowych – związanych z realizacją rozkładu jazdy i osiągnięciem punktów zgłaszania priorytetu. ZDIUM Wrocław dla każdego pojazdu będzie wskazywać jaki wariant zgłaszania priorytetu musi być realizowany przez pojazd.

Dostawca danych musi być przygotowany do realizacji zgłaszania priorytetu w każdym wariantcie opisanym szczegółowo w rozdziale 2.3 niniejszego dokumentu.

Serwer aplikacyjny został skonfigurowany do nasłuchiwania połączeń HTTPS (port 8443), dla udostępnionych zasobów REST. Wszystkie zasoby wymagają przedstawiania się właściwym certyfikatem klienckim SSL. Wyjątkiem od tej zasady jest transmisja w prywatnym APN ITS, która nie wymaga przedstawienia się certyfikatem SSL (w tym przypadku karty SIM zostaną dostarczone przez ZDIUM).

Serwer MIT jest zlokalizowany pod adresem <https://mit.its.wroc.pl:8443>

Interfejs API może zwrócić błędy w przypadku błędnych zapytań do serwera:

- HTTP 403 Forbidden – brak autoryzacji
- HTTP 404 Bad Request – błędny adres zapytania
- HTTP 500 Internal Server Error – błąd wewnętrzny serwera, związany z operacjami na danych lub błędzie w połączeniu do bazy danych

Ogólna postać zwracanego błędu:

```
{  
  "timestamp": [data czas – znacznik czasowy wystąpienia błędu],  
  "status": [liczba całkowita – kod błędu HTTP],  
  "error": [tekst – nazwa błędu HTTP],  
  "exception": [tekst – nazwa klasy wyjątku logowanego po stronie serwera],  
  "message": [tekst – treść wyjątku],  
  "path": [tekst – adres zapytania]  
}
```

Wszystkie dane zwracane są w postaci: *application/json; charset=UTF-8*. Data i czas reprezentowany jest formacie ISO *yyyy-MM-dd'T'HH:mm:ssZ* zarówno w zwracanych błędach jak i wszystkich pozostałych odpowiedziach i zapytaniach do serwera.

2.2 Konfiguracja nowych dostawców danych po stronie systemu ITS

Dane z pojazdów mogą być dostarczane zarówno w architekturze serwer-serwer (w przypadku, gdy Dostawca danych dysponuje własnym serwerem przetwarzającym dane z urządzeń) lub pojazd-serwer (interfejs może być zaimplementowany po stronie komputera pokładowego).

2.2.1 Konfiguracja po stronie systemu ITS dla komunikacji serwer-serwer

Dla komunikacji serwer-serwer konieczna jest autoryzacja przy pomocy certyfikatu SSL.

Po stronie systemu ITS - nowy użytkownik (dostawca danych), musi zostać wprowadzony przez Wykonawcę ITS.

Serwer przewoźnika musi przedstawiać się wygenerowanym dla niego certyfikatem klienckim, zaufanym przez serwer MIT. Dostawca danych, którego certyfikat nie będzie zaufany przez serwer aplikacyjny, nie zostanie upoważniony do pobierania/wysyłania danych.

Dla nowych dostawców danych zostanie przesłany wygenerowany certyfikat do komunikacji https wraz z krótką instrukcją wykorzystania tego certyfikatu do komunikacji z serwerem MIT (rozdział 5).

2.2.1.1 Dodawanie nowego przewoźnika oraz generowanie certyfikatu dla przewoźnika

Generowanie certyfikatu odbywa się za pomocą skryptu *keystores.bat*, który wymaga narzędzia *keytool*.

Dostawca danych musi zostać wcześniej umieszczony przez Wykonawcę w tabeli *MIT_DATA_SOURCES*.

Wymagane są również pliki:

server.jks – prywatny keystore/truststore serwera, znajdujący się w lokalizacji
/opt/tomcat/conf/ssl/server

server.crt – publiczny klucz certyfikatu serwera (znajduje się w pliku server.jks i jest możliwy do wyciągnięcia wygodnie np. za pomocą programu KeyStoreExplorer)

ca.crt – publiczny klucz certyfikatu CA reprezentującego wydawcę – ZDIUM (znajduje się w pliku server.jks i jest możliwy do wyciągnięcia wygodnie np. za pomocą programu KeyStoreExplorer)

Należy umieścić powyższe pliki w tym samym folderze, co skrypt keystores.bat.

Wykonać: keystores.bat -c <id> <password> <organization> <city> <state>

gdzie:

<id> - id bazodanowy przewoźnika wygenerowany podczas umieszczenia obiektu przewoźnika w bazie danych

<password> - hasło do wynikowych plików .jks oraz .p12

<organization>, <city>, <state> - dane podmiotu, dla którego generowany jest certyfikat (nazwa, miasto, województwo)

Jako wynik działania skryptu powstaną pliki „id_provider_<id>.jks” oraz „id_provider_<id>.p12” zabezpieczone podanym hasłem. Certyfikat ten zawiera klucz prywatny i publiczny klienta MIT, zaufany przez certyfikat serwera MIT.

Zostanie również zaktualizowany plik server.jks, w którym zostanie umieszczony nowy wpis zaufanego klienta.

2.2.2 Konfiguracja dla komunikacji pojazd-serwer

W przypadku komunikacji pojazd-serwer autoryzacja użytkowników w systemie odbywa się na podstawie weryfikacji zgodności adresu karty GSM podłączonej do dedykowanego prywatnego APN dostarczonej przez ZDIUM.

Dla takiej komunikacji będą udostępnione takie same metody w interfejsie komunikacyjnym, ale nie będzie wymagana autoryzacja za pomocą certyfikatu SSL.

Zakres adresacji IP dla komputerów pokładowych pracujących w prywatnym APN zostanie uzgodniony z Zamawiającym.

2.3 Definicja wariantów związanych z przesyłaniem punktów zgłoszeniowych do systemu ITS

Dostawca danych do systemu ITS może dostarczać do systemu ITS zgłoszenia priorytetu w 3 wariantach:

- **Wariant A** – Wariant zgłaszania priorytetu przejazdu dla linii tramwajowej. Dane wejściowe stanowią ramki lokalizacyjne przesyłane co 15 sekund oraz zgłoszenia priorytetu, które są powiązane z segmentem trasy (parą słupków) rozkładu jazdy. W definicji punktów zgłoszeniowych znajduje się również konfiguracja modułów radiowych na

skrzyżowaniach, do których zdarzenie osiągnięcia przez pojazd punktu zgłoszeniowego musi być wysłane również przy pomocy radia krótkiego zasięgu.

- **Wariant B** – Wariant zgłaszania priorytetu przejazdu (autobus lub tramwaj) dla pojazdów posiadających komputery pokładowe zasilone wirtualnymi punktami zgłoszeniowymi niezależnymi od rozkładów jazdy. Pojazdy niezależnie od realizowanej trasy zgłaszają żądanie priorytetu w każdym obszarze zdefiniowanym w tym wariantcie.
- **Wariant C** – Wariant zgłaszania priorytetu przejazdu (autobus) linii przejazdu linii priorytetowej. Komputery pokładowe zmieniają częstość nadawania ramek synchronicznych do systemu ITS z 15s na 2s (jeśli znajdują się wewnątrz zdefiniowanych obszarów punktów zgłoszeniowych).

W zależności od potrzeb właściciel systemu ITS – ZDIUM Wrocław – będzie informował dostawcę danych o terminie wprowadzenia nowego sposobu zgłaszania priorytetu przez wskazane pojazdy (z wyprzedzeniem 14 dniowym).

Dostawca danych musi mieć możliwość rekonfiguracji komputerów pokładowych we własnych pojazdach oraz być przygotowany do realizacji zgłaszania priorytetu w każdym wariantcie opisanym w niniejszym dokumencie.

2.4 Konfiguracja repozytorium danych rozkładowych

Każdy dostawca danych zobowiązany jest do realizacji zadań przewozowych wynikających z umowy. W poniższym interfejsie dostawca danych zobowiązany jest do przesyłania ramek asynchronicznych związanych z realizacją rozkładu jazdy.

Organizatorem transportu publicznego we Wrocławiu jest Wydział Transportu Urzędu Miasta (WTR). WTR przygotowuje rozkłady jazdy dla wszystkich przewoźników (dostawców danych) w oprogramowaniu Busman. Wszystkie zatwierdzone rozkłady jazdy trafiają do bazy danych Busman CB, która jest bazą pozwalającą na integrację z innymi systemami.

Wszystkie identyfikatory przesyłane w rekordach danych muszą być zgodne z identyfikatorami w bazie Busman CB, której właścicielem jest WTR.

3. Interfejs wymiany danych dla pojazdów transportu publicznego

3.1 Opis zasobów dostawców danych

Zasoby dostawców danych pozwalają zweryfikować skonfigurowanych w bazie danych dostawców oraz sprawdzić poprawność klienckiego certyfikatu SSL.

Model danych obiektu dostawcy:

```
{  
  "id": [liczba całkowita – identyfikator bazodanowy],  
  "name": [tekst – nazwa dostawcy]  
}
```

- Adres: <https://mit.its.wroc.pl:8443/api/MIT/providers>
Metoda: GET
Odpowiedź: HTTP 200 - tablica obiektów dostawców danych
Dostęp ograniczony wyłącznie dla klienta administracyjnego
- Adres: <https://mit.its.wroc.pl:8443/api/MIT/providers/current>
Metoda: GET
Odpowiedź: HTTP 200 - własny obiekt dostawcy danych bazujący na klienckim certyfikacie SSL

3.2 Opis zasobów zdarzeń

Zasoby zdarzeń zwracają listę możliwych typów zdarzeń, które mają być wysyłane przez dostawców danych. Końcówka API utworzona w celu statycznej weryfikacji i upraszczająca integrację.

Model danych obiektu zdarzeń:

```
{  
  "id": [liczba całkowita – identyfikator bazodanowy],  
  "description": [tekst – opis zdarzenia]  
}
```

Adres: <https://mit.its.wroc.pl:8443/api/MIT/events>

Metoda: GET

Odpowiedź: HTTP 200 - tablica obiektów zdarzeń

Oczekiwana lista zdarzeń wymagana do integracji z systemem ITS (identyfikatory zgodne z obecną numeracją):

Identyfikator	Nazwa	Opis
0	Lokalizacja GPS	Zdarzenie <i>Lokalizacja GPS</i> jest zdarzeniem synchronicznym, generowanym przez urządzenie pokładowe z określonym interwałem czasu ustawionym na urządzeniu Uwaga: Wymagany maksymalny interwał czasowy w strefach detekcji wariantu C wynosi 2 sekundy, dla pozostałych lokalizacji 15 sekund.
11	Zalogowanie kursu / Wylogowanie kursu	Zdarzenie <i>Zalogowanie/wylogowanie kursu</i> jest zdarzeniem asynchronicznym informującym o wybraniu przez motorniczego kursu, który będzie realizowany lub rezygnacji z realizacji wybranego uprzednio kursu (negacja wartości w polu course oznacza wylogowanie z tego kursu)
12	Rozpoczęcie kursu	Zdarzenie <i>Rozpoczęcie kursu</i> jest zdarzeniem asynchronicznym informującym o rozpoczęciu realizacji kursu.
13	Zakończenie / przerwania kursu	Zdarzenie <i>Przerwanie kursu</i> jest zdarzeniem asynchronicznym informującym o ukończeniu realizacji kursu.
14	Przyjazd na przystanek	Zdarzenie <i>Przyjazd na przystanek</i> jest zdarzeniem asynchronicznym informującym o wjeździe pojazdu w strefę przystanku i zatrzymaniu pojazdu.

15	Otwarcie drzwi	Zdarzenie <i>Otwarcie drzwi</i> jest zdarzeniem asynchronicznym informującym o otwarciu drzwi w strefie przystanku.
16	Zamknięcie drzwi	Zdarzenie <i>Zamknięcie drzwi</i> jest zdarzeniem asynchronicznym informującym o zamknięciu drzwi w strefie przystanku.
17	Odjazd z przystanku	Zdarzenie <i>Odjazd z przystanku</i> jest zdarzeniem asynchronicznym informującym o ruszeniu pojazdu i opuszczeniu strefy przystanku.
18	Przejazd przez przystanek bez zatrzymania	Zdarzenie <i>Przejazd przez przystanek bez zatrzymania</i> jest zdarzeniem asynchronicznym informującym o sytuacji, w której pojazd przejechał przez strefę przystanku zaplanowanego na realizowanej trasie bez zatrzymywania się na nim.
22	Dojazd do zajezdni	Zdarzenie <i>Dojazd do zajezdni</i> jest zdarzeniem asynchronicznym informującym o wjeździe pojazdu na teren zajezdni.
23	Wyjazd z zajezdni	Zdarzenie <i>Wyjazd z zajezdni</i> jest zdarzeniem asynchronicznym informującym o wyjeździe pojazdu z terenu zajezdni.
101	Zdarzenie związane ze strefą detekcji	Zdarzenie jest zdarzeniem asynchronicznym informującym o wjeździe lub wyjeździe pojazdu do/z obszaru detekcji (w zależności od wartości parametru notificationType)

3.3 Opis zasobów obszarów detekcji

Metoda pozwala na pobranie listy punktów detekcji, wykorzystywanych przez system ITS do celów sterowania ruchem oraz wyznaczania predykcji czasów przejazdu na tablicach DIP. Zasoby te definiują strefy detekcji GPS, które w zależności od typu takiej strefy powinny być w odpowiedni sposób interpretowane przez komputer pokładowy. W poniższej tabeli opisano wszystkie typy takich stref oraz wymagania dla komputerów pokładowych, które muszą w odpowiedni sposób zareagować znajdując się w takiej strefie.

Pole „detectionType” może przyjmować następujące wartości:

detectionType	Nazwa	Opis
1	Strefa przystankowa	Komputer pokładowy będąc wewnątrz tej strefy musi wysłać dodatkowe zdarzenia asynchroniczne 14-18 (otwarcie/zamknięcie drzwi, przyjazd na przystanek, opuszczenie przystanku), pozwalające na przekazanie odpowiedniego sygnału do systemu sterowania ruchem.
2	wariant A	Komputer pokładowy, który realizuje zgłaszanie priorytetu zgodnie z wariantem A musi przesłać do systemu ITS komunikat 101. Żądanie jest wysyłane wyłącznie gdy pojazd realizuje kurs pomiędzy przystankami zdefiniowanymi jako postIdFrom, postIdTo. Równolegle musi zostać wysłany komunikat do odbiornika radiowego na skrzyżowaniu (definicja parametrów dla radia zawiera się w parametrach onlyWariantA).
3	wariant B	Komputer pokładowy, który realizuje zgłaszanie priorytetu zgodnie z wariantem B musi przesłać do systemu ITS komunikat 101. Równolegle musi zostać wysłany komunikat do odbiornika radiowego, jeśli dla tego detektora zostały

		uzupełnione pola RFAAddress, RFChannel
4	wariant C	Komputery pokładowe, który realizuje obsługę priorytetu zgodnie z wariantem C po wjechaniu do takiej strefy detekcji musi zmienić częstotliwość przesyłania ramek lokalizacyjnych do systemu ITS z 15s na 2s. Po wyjechaniu ze strefy częstotliwość przesyłania ramek powinna przywrócona do 15s.

Model danych obiektu obszaru detekcji:

```
{
  "id": [liczba całkowita – identyfikator bazodanowy],
  "name": [tekst – nazwa opisowa detektora],
  "detectionType": [liczba całkowita - zgodnie powyższą tabelą],
  "postSymbol": [tekst – symbol słupka dla definicji stref przystankowych],

  "notificationType": [liczba całkowita – rodzaj oczekiwanego powiadomienia. Pole notificationType
    reprezentuje bitowo następujące rodzaje notyfikacji:
    1b ( $2^0$ ) – wyjazd z obszaru,
    2b ( $2^1$ ) – wjazd do obszaru,
  "latitude": [liczba zmiennoprzecinkowa – szerokość geograficzna w odniesieniu WGS84],
  "longitude": [liczba zmiennoprzecinkowa – długość geograficzna w odniesieniu WGS84],
  "radius": [liczba całkowita – promień obszaru wyrażony w metrach],
```

onlyWariantA:

```
{
  "RFAAddress": [liczba całkowita - opcjonalnie adres radia krótkiego zasięgu],
  "RFChannel": [liczba całkowita - opcjonalnie kanał radia krótkiego zasięgu],
  "postIdFrom": [liczba całkowita - identyfikator słupka segmentu trasy, pozwalający na
    powiązanie detektora z unikalnym segmentem trasy w Busman],
  "postIdTo": [liczba całkowita - identyfikator słupka segmentu trasy, pozwalający na powiązanie
    detektora z unikalnym segmentem trasy w Busman],
  "entryNo": [liczba całkowita - opcjonalnie numer zgłoszenia dla systemu sterowania],
  "distanseFromPost": [liczba całkowita - oznacza odległość od początkowego słupka segmentu
    trasy],
```

```

    "iDirection": [liczba całkowita - kierunek przejazdu przez skrzyżowanie],
    "exceptions": [lista numerów kursów, dla których nie będą wysyłane żądania priorytetu],
    {
        "courseID": [liczba całkowita]
    }
}
}

```

Adres: <https://mit.its.wroc.pl:8443/api/MIT/area/detections>

Metoda: GET

Odpowiedź: HTTP 200 – tablica obiektów obszarów detekcji

3.4 Opis zasobów rekordów danych

Zasoby rekordów danych pozwalają dostarczyć dane od dostawców danych (przewoźników) do systemu ITS.

Jest to jedyna metoda POST, a więc ta metoda pozwala na przekazanie ramek synchronicznych oraz asynchronicznych do systemu ITS.

Model danych rekordów:

```

{
    "id": [liczba całkowita – unikalny identyfikator bazodanowy lub numer sekwencji pozwalający na
        kontrolę kompletności lub kolejności ramek],
    "idProvider": [liczba całkowita – identyfikator dostawcy (zgodny z providers.id)],
    "idEvent": [liczba całkowita – identyfikator zdarzenia (zgodny z events.id)],
    "sideNumber": [tekst – numer boczny pojazdu],
    "plateNumber": [tekst – tablica rejestracyjna do 8 znaków],
    "timeRegistered": [data czas – czas rejestracji na urządzeniu pokładowym],
    "timeSent": [data czas – czas rejestracji w systemie],
    "latitude": [liczba zmiennoprzecinkowa – szerokość geograficzna w odniesieniu WGS84],
    "longitude": [liczba zmiennoprzecinkowa – długość geograficzna w odniesieniu WGS84],
    "velocity": [liczba całkowita – prędkość chwilowa wyrażona w m/h],
    "tacho": [liczba całkowita – wskazanie tachometru pojazdu],
    "vehicleType": [liczba całkowita – wyróżnik rodzaju pojazdu (1 – autobus, 2 – tramwaj)],
    "lineName": [tekst – symbol obsługiwanej linii],
    "busmanId": [liczba całkowita – identyfikator rozkładu jazdy załadowanego w komputerze
        pokładowym, identyfikator z bazy BusmanCB – trozklady.id],

```

```

"idRouteVariant": [liczba całkowita – identyfikator wariantu trasy, identyfikator z bazy BusmanCB
                  – twarianty.id],
"brigade": [tekst – nazwa brygady, identyfikator z bazy BusmanCB – tzadania.nazwa],
"course": [liczba całkowita – identyfikator kursu realizowanego przez pojazd (identyfikator z
          Busman CB – tkursy.id)],
"delay": [liczba całkowita – opóźnienie (dodatnia) lub nadspieszenie (ujemna) na ostatnim
          odwiedzionym słupku przystankowym],
"postSymbol": [tekst – symbol słupka],
"driver": [tekst – identyfikator kierowcy, np. numer służbowy],
"detectionId": [liczba całkowita – identyfikator strefy GPS zgodny z detections.id)],
„notificationType”: [liczba całkowita – rodzaj wyzwalacza detekcji, analogicznie do definicji strefy
                     GPS, pole notificationType reprezentuje bitowo następujące rodzaje notyfikacji:
                     1b ( $2^0$ ) – wyjazd z obszaru,
                     2b ( $2^1$ ) – wjazd do obszaru,
                     }

```

Adres: <https://mit.its.wroc.pl:8443/api/MIT/records>
 Metoda: POST
 Nagłówek: *application/json; charset=UTF-8*
 Body: tablica obiektów rekordów
 Zwraca: HTTP 204 No Content

Wartości wymagane dla wszystkich typów zdarzeń (idEvent):

```

{
    id
    idProvider
    idEvent
    sideNumber
    timeRegistered
    latitude
    longitude
    velocity
    tachometer
    course //również w ramce lokalizacyjnej – jeśli w tym czasie jest realizowana praca
           przewozowa
}

```

Dodatkowo dla pozostałych typów zdarzeń wymagane są następujące parametry:

idEvent		
11	Zalogowanie/wylogowanie kursu	busmanId idRouteVariant brigade
12	Rozpoczęcie kursu	course
13	Zakończenie kursu	driver
14	Zatrzymanie na przystanku	busmanId idRouteVariant brigade
15	Otwarcie drzwi	course
16	Zamknięcie drzwi	delay
17	Odjazd z przystanku	postSymbol driver
18	Przejazd bez zatrzymania	
22	Dojazd do zajezdni	busmanId
23	Wyjazd z zajezdni	driver
101	Zdarzenie związane ze strefą detekcji	detectionId notificationType

4. Dokumentacja API

Opis wszystkich metod dostępnych w ramach usługi wraz z przykładowymi zapytaniami i odpowiedziami dostępny jest pod adresem <http://mit.its.wroc.pl:8443/swagger-ui.html>

Do wyświetlenia tej strony w oknie przeglądarki konieczne jest dodanie klienckiego certyfikatu SSL do przeglądarki, zaufanego przez serwer aplikacyjny.

W załączniku do niniejszego dokumentu znajduje się dokumentacja wygenerowana za pomocą oprogramowania swagger.



Swagger_MIT.pdf

5. Przykładowa implementacja konfiguracji klienta https

Przykładowa implementacja konfiguracji klienta https dla połączenia z serwerem MIT przy użyciu certyfikatu klienckiego SSL w języku Java, przy użyciu biblioteki Apache HttpComponents:

```
String password = "...";
String jksPath = "...";

SSLContext sslContext = SSLContexts.custom()
    .loadTrustMaterial(
        new File(jksPath),
        password.toCharArray()
    )
    .loadKeyMaterial(
        new File(jksPath),
        password.toCharArray(),
        password.toCharArray()
    )
    .build();

HttpClient client = HttpClientBuilder.create()
    .setSSLContext(sslContext)
    .build();
```

Zmienna jksPath powinna wskazywać na lokalizację przekazanego pliku „id_provider_<id>.jks”, a zmienna password przechowuje hasło do przekazanego pliku (hasło jest takie samo zarówno dla magazynu jks, jak i klucza prywatnego).

Utworzona instancja interfejsu HttpClient może zostać następnie użyta np. do zbudowania obiektu RestTemplate znanego frameworka Spring:

```
HttpClientBuilder builder = new
    HttpClientBuilder()
    .setSSLContext(sslContext);

RestTemplate template = new RestTemplate(builder);
```